

# スクラッチ ドリルブック

作って学ぶ実践プログラミング練習帳

石原 正雄 ● 著

HAUNT WORKS STUDIO / 原本優莉 ● 協力



## この冊子の使い方

この冊子は「スクラッチドリルブック / 解答例（サンプルプログラム集）」です。

この冊子はカットシステムから出版された「スクラッチ ドリルブック」に掲載されたチャレンジ（課題）の解答例やサンプルプログラムを掲載し、合わせて正誤情報や出版時からスクラッチソフトウェア、特に 3.0 ベータバージョンが変更された内容について説明してあります。

「スクラッチ ドリルブック」は「チャレンジ」と呼ばれるまとまりで関連する命令やプログラミングの考え方についての演習課題を紹介しています。各チャレンジはどれも同じ構成です。

まずそのチャレンジで取り上げる課題についての説明があります。続いて下の3つのセクションがあります。

- ・コードペディア：プログラムの一部を隠したり、未完成にしてあります。  
自分で考えて完成させることで基本的な命令の使い方やプログラミングの考え方を学びます。
- ・ヒント：コードペディアに関係する手がかりを示します。
- ・発展チャレンジ：応用課題です。

この冊子では、上記のコードペディア、ヒント、発展チャレンジのそれぞれで完成したプログラムのサンプルを掲載しています。この冊子のサンプルプログラムは必ず自分でも実行してみて、なぜそのような動きになるのかを考えるようにしてください。そして、さらに面白くするにはどうしたらよいかアイデアをいろいろ考えて、サンプルプログラムをどんどん発展させてみてください。

2018年 10月 3日 著者

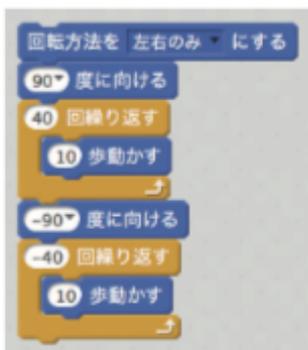
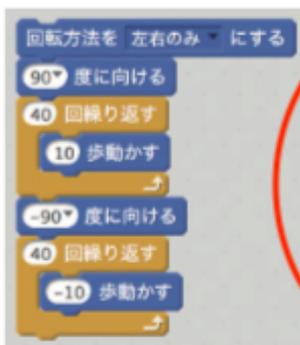
# チャレンジ 01

コードペディア（解答例）



ヒント

ヒントの中の3つのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。ただし、右から左へ動くのが速くなっています。行き（左から右）、帰り（右から左）で速さを同じにしたい時は上図のようなプログラムにします。



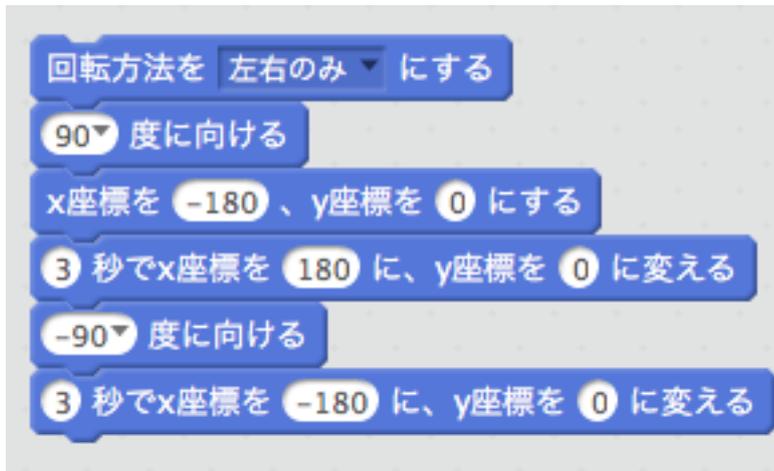
## 発展チャレンジ

### サンプルプログラム



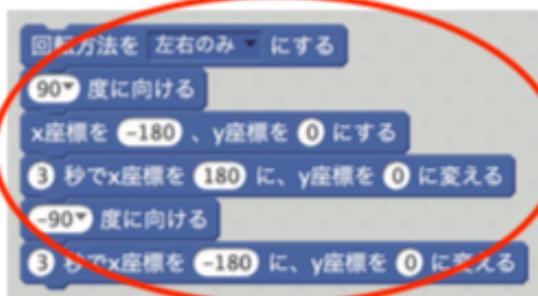
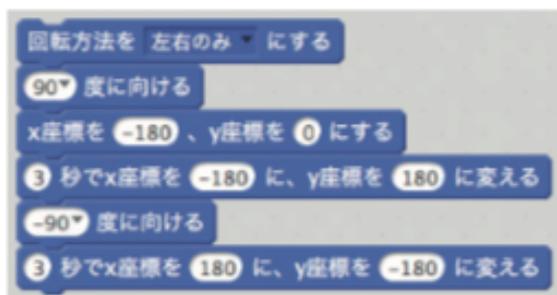
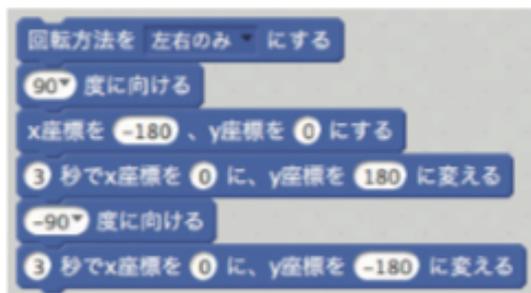
## チャレンジ 02

コードペディア (解答例)



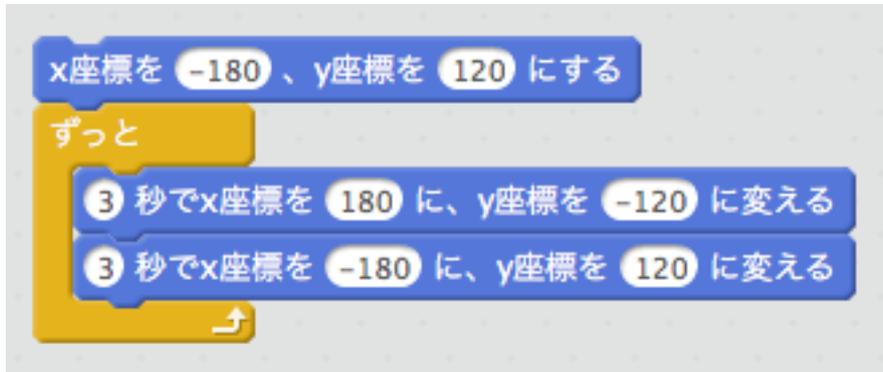
ヒント

ヒントの中の3つのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



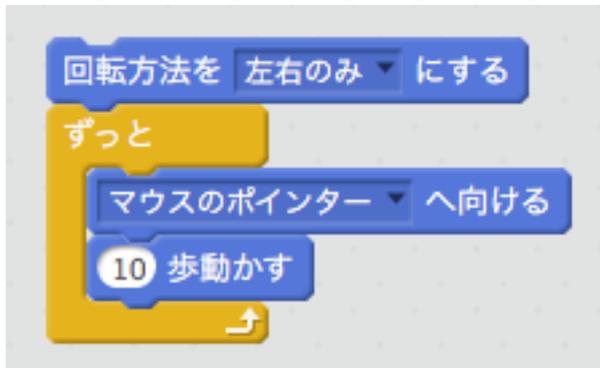
## 発展チャレンジ

### サンプルプログラム



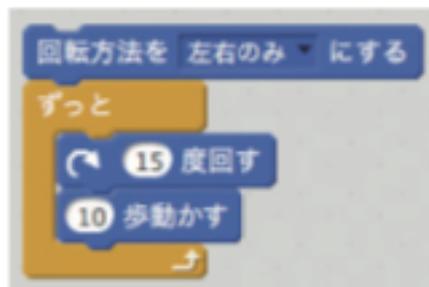
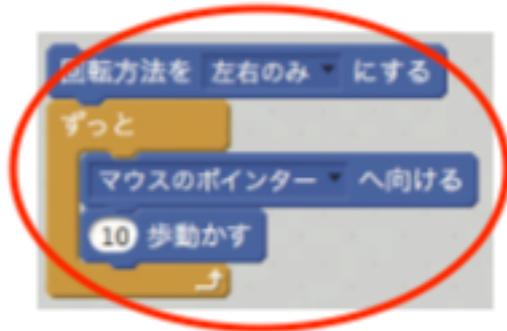
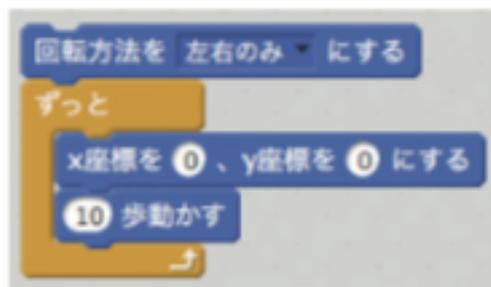
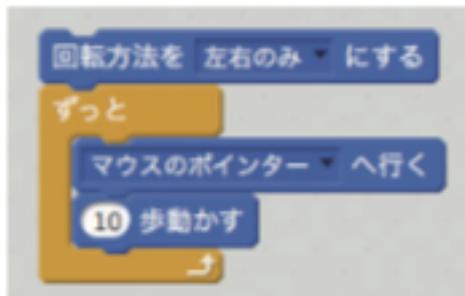
## チャレンジ 03

コードペディア（解答例）



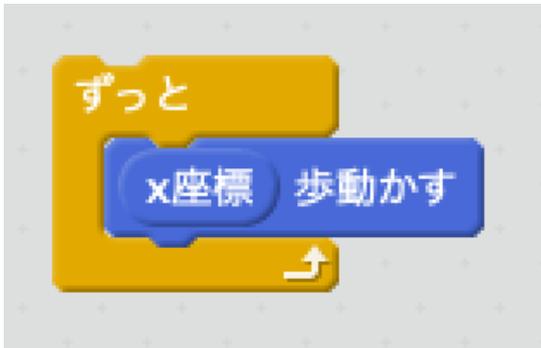
ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



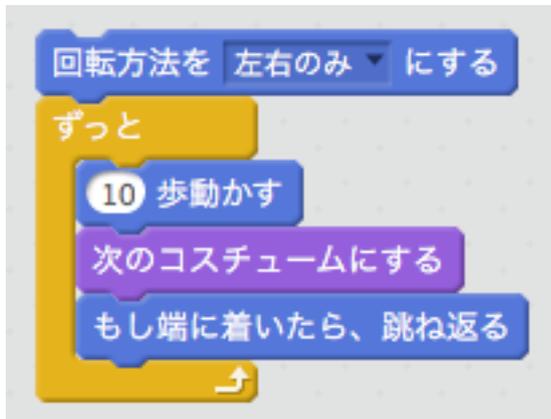
発展チャレンジ

サンプルプログラム



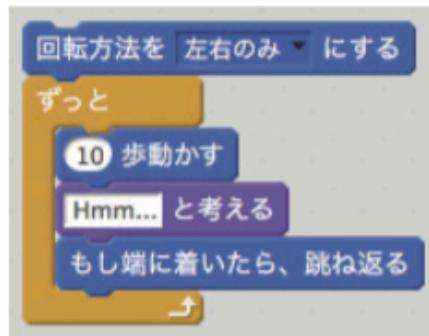
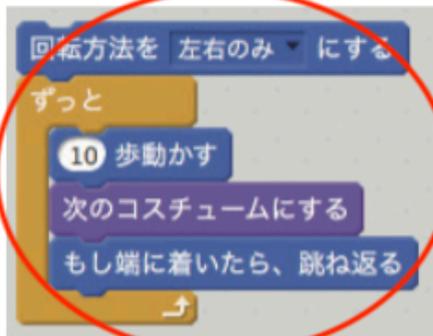
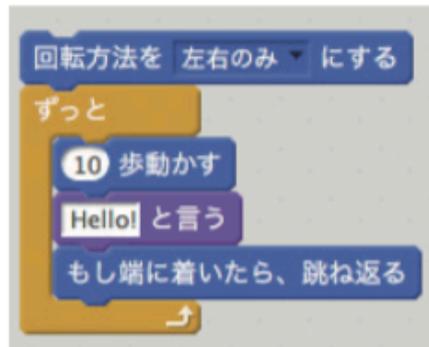
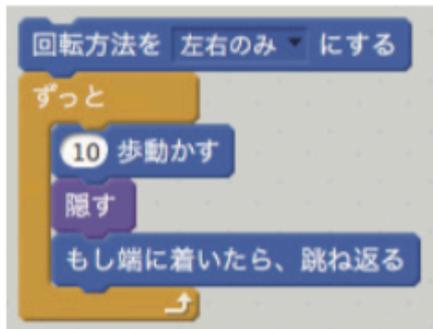
## チャレンジ 04

コードペディア (解答例)



ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



## 発展チャレンジ

- ・準備として2つの背景を用意しましょう。

以下は背景を加えるやり方の説明です。

- 1) 画面左下の「ステージ」をクリックする。



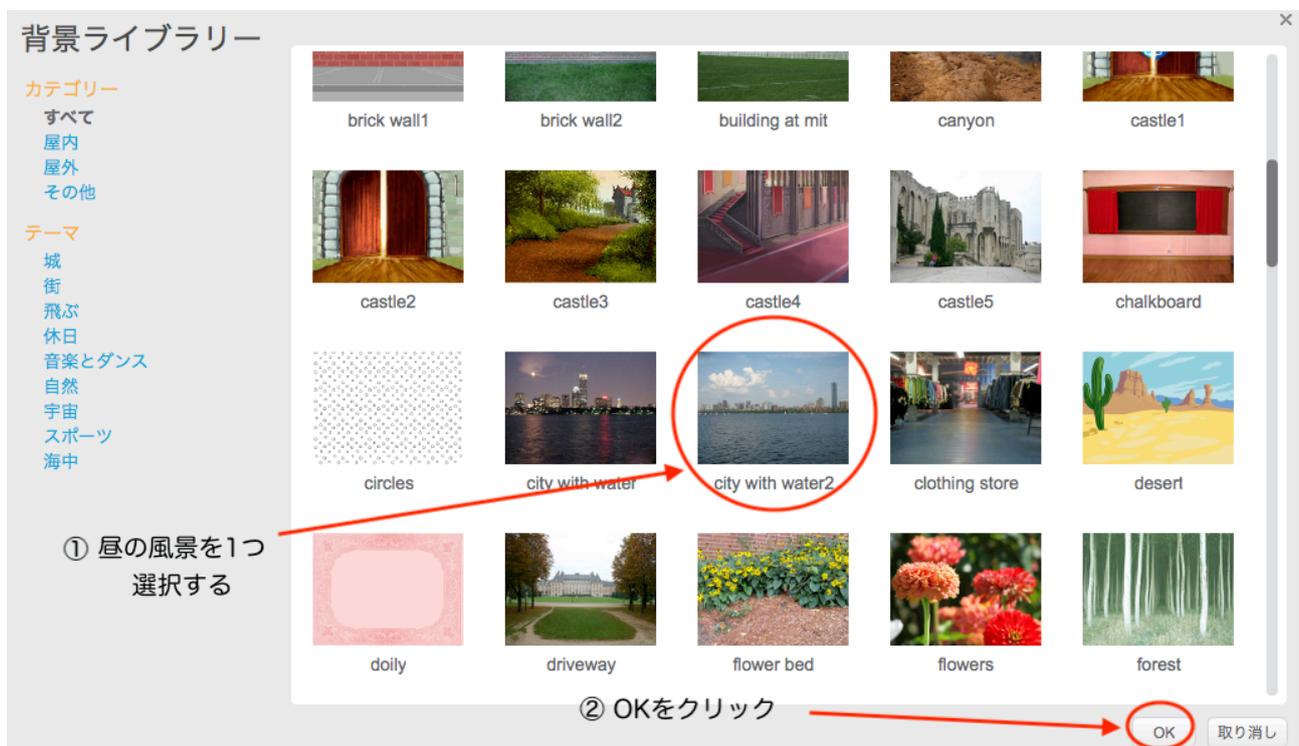
- 2) 「背景」タブをクリックする。



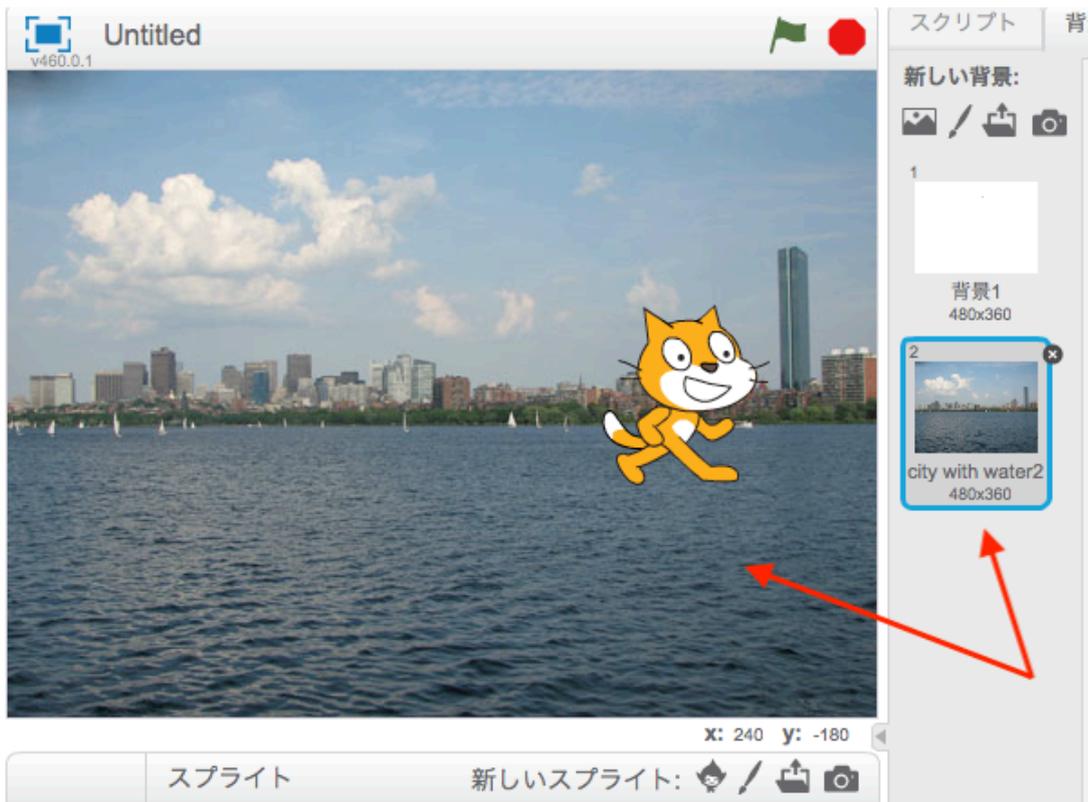
3) 「新しい背景：」の下のアイコンから一番左をクリック。「ライブラリーから背景を選択」と表示されます。



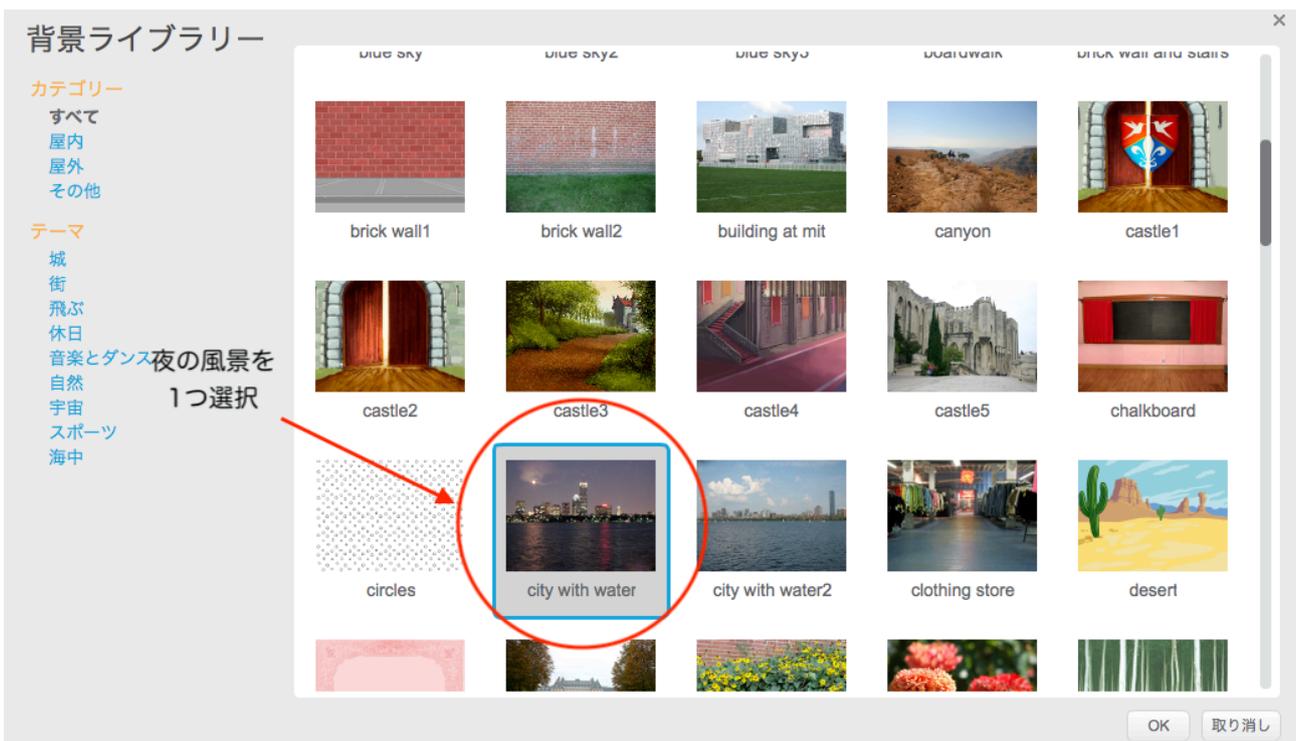
4) 「背景ライブラリー」が表示されたら昼の風景を1つ選択する（下図の例では city with water 2）。選択したら「OK」をクリックする。



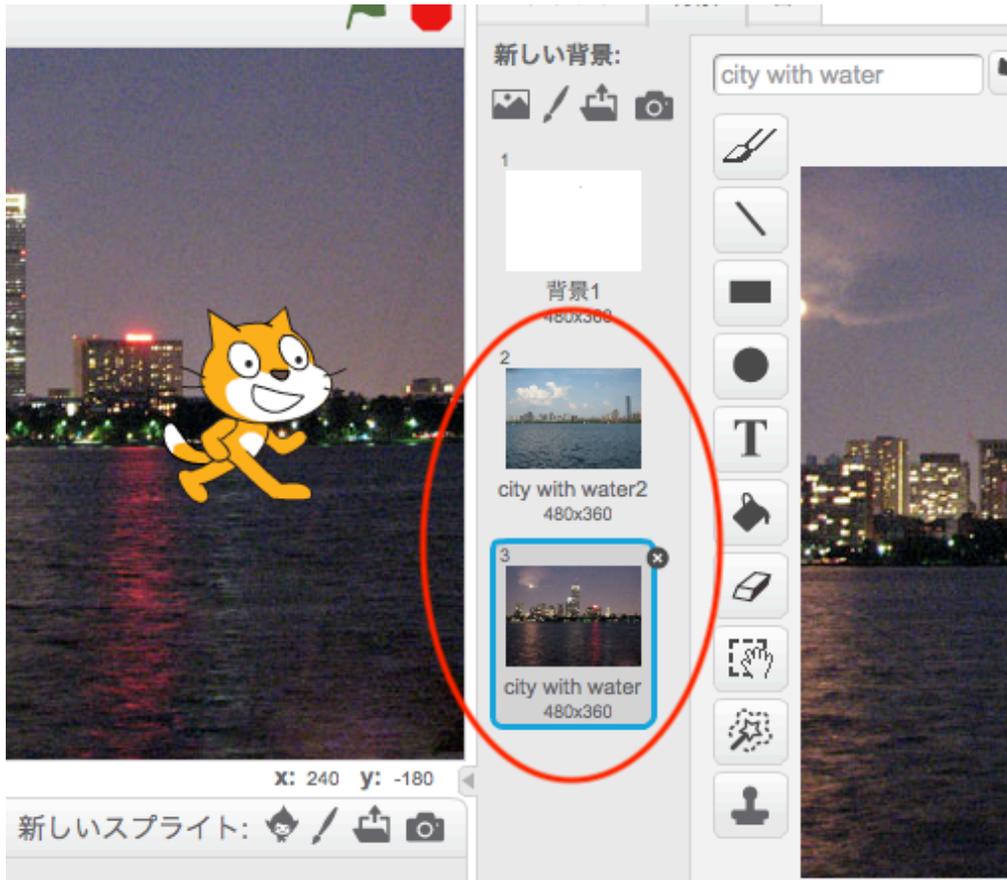
5) 下図のように新しい背景が追加される。



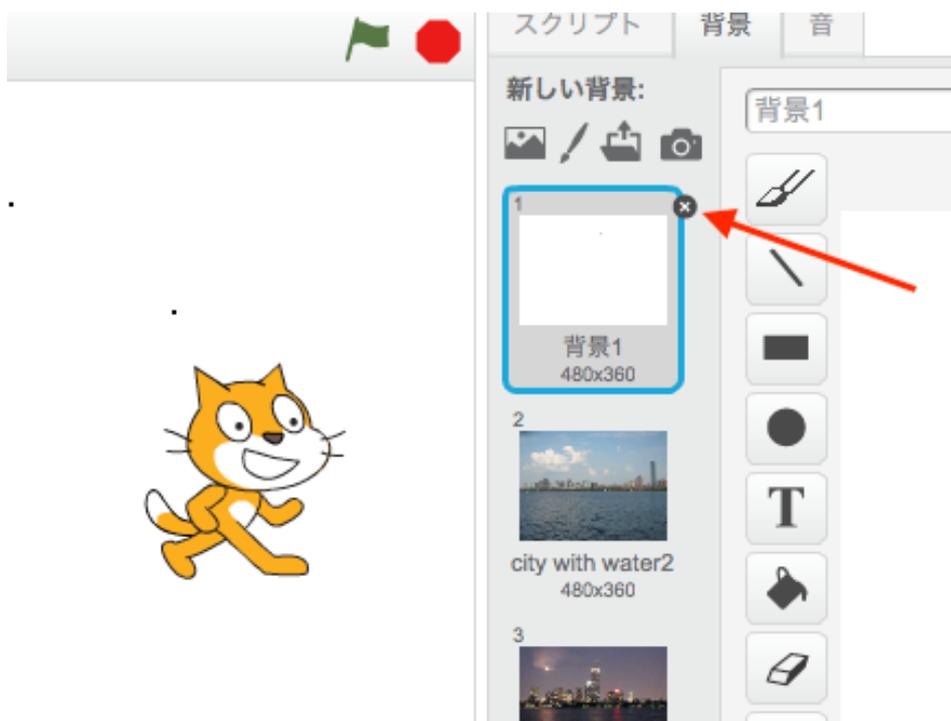
6) 上の3) と4) の手順を繰り返して夜の風景を追加する。



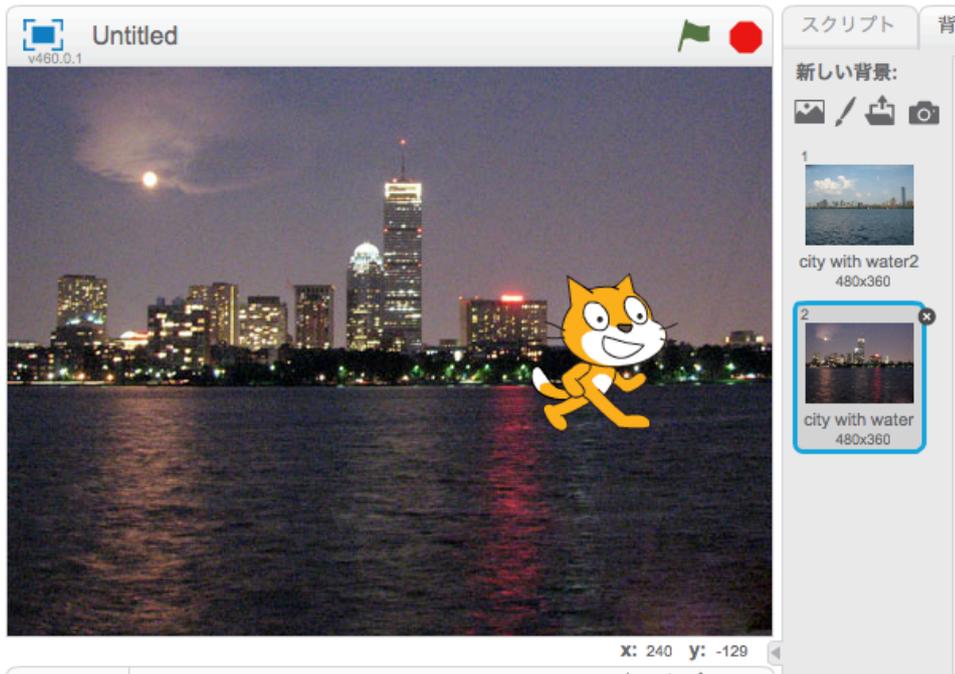
7) 昼と夜、2つの風景が追加されました。



8) 最初にあった白地の背景は不要なので下図の矢印が示す x 印 をクリックして削除する。



9) 下図のように昼と夜の2つの背景が追加されて準備ができました。



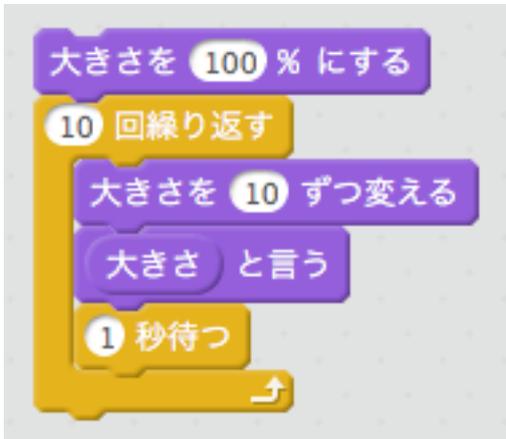
サンプルプログラム

下図はサンプルプログラムです。



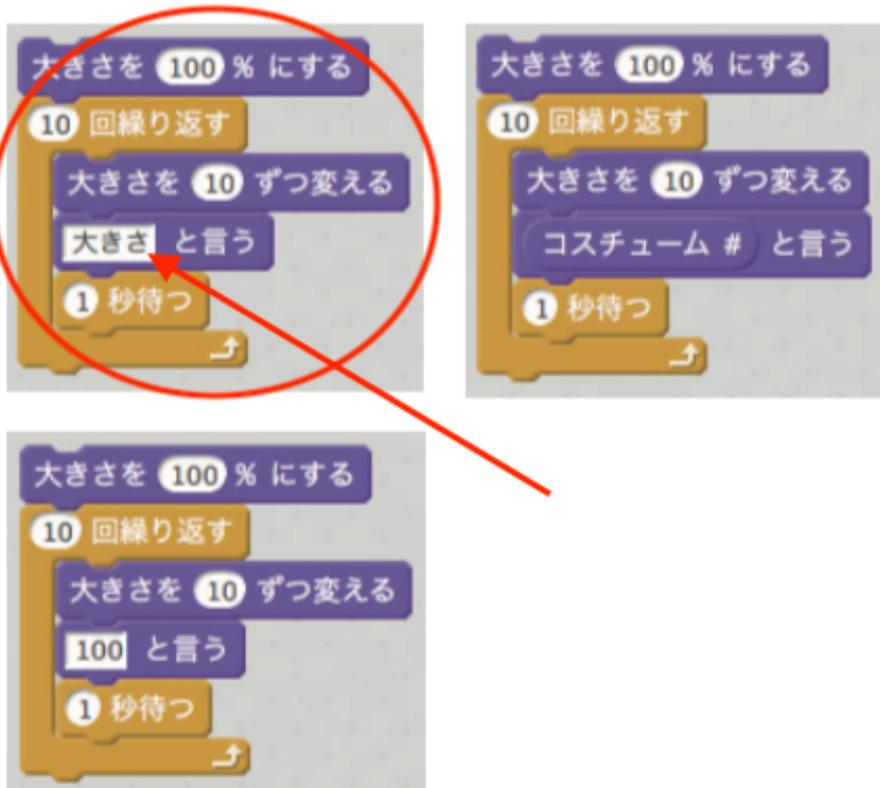
## チャレンジ 05

コードペディア（解答例）



ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムで動きそうですが、赤矢印の先にある「大きさ」はタイプ入力するのではなくて「見た目」カテゴリーにある「大きさ」を使う必要があります。完成例は上図のコードペディア（解答例）のようになります。



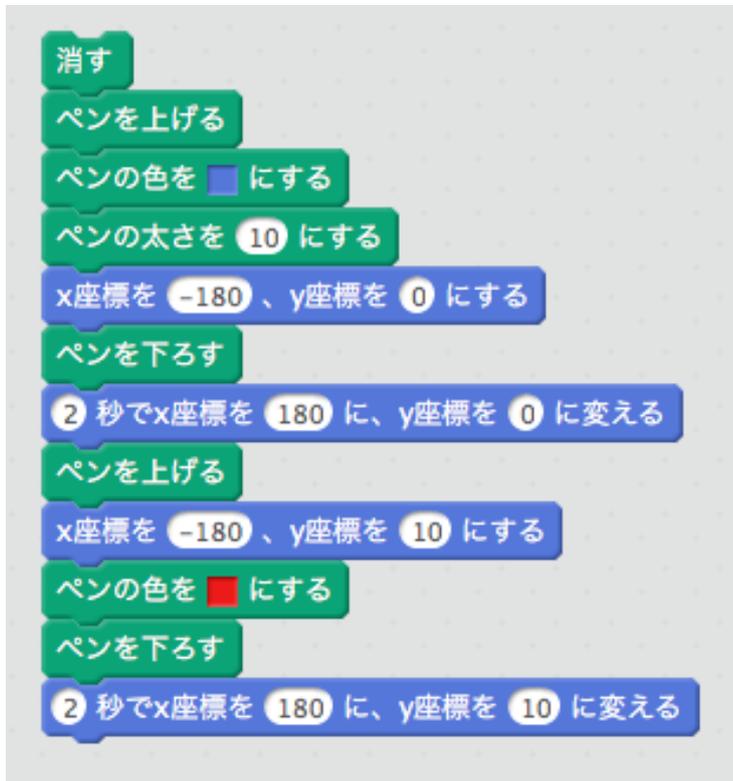
## 発展チャレンジ

### サンプルプログラム

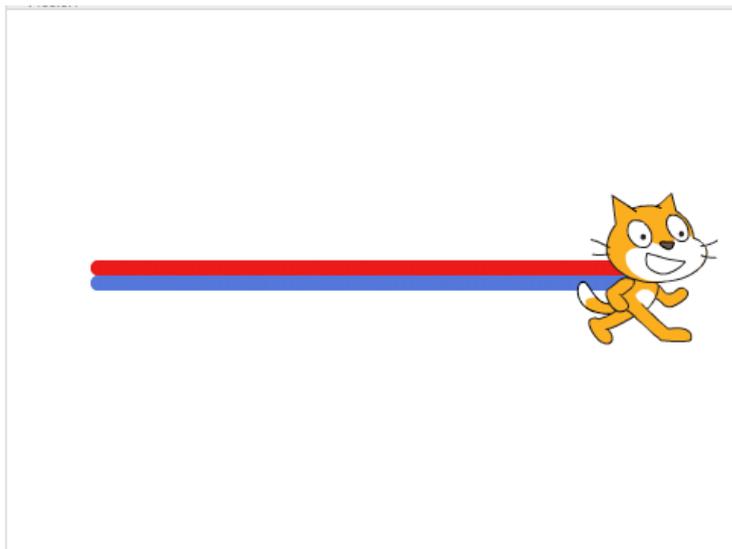


## チャレンジ 06

コードペディア（解答例）

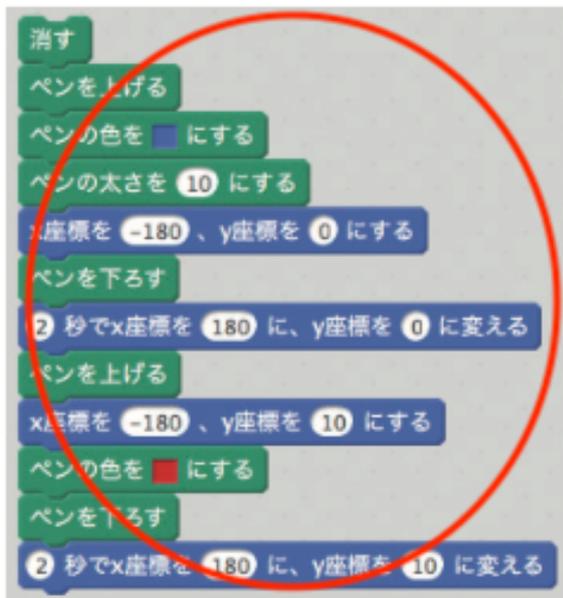
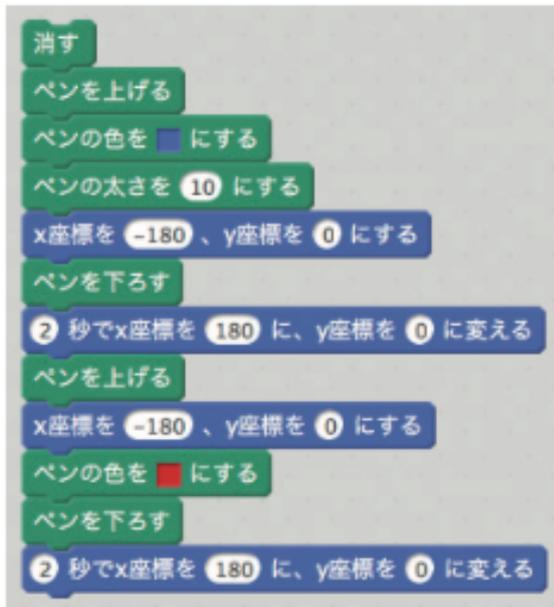


上図のプログラムを実行すると下図のように2本の線が描けます。



## ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。

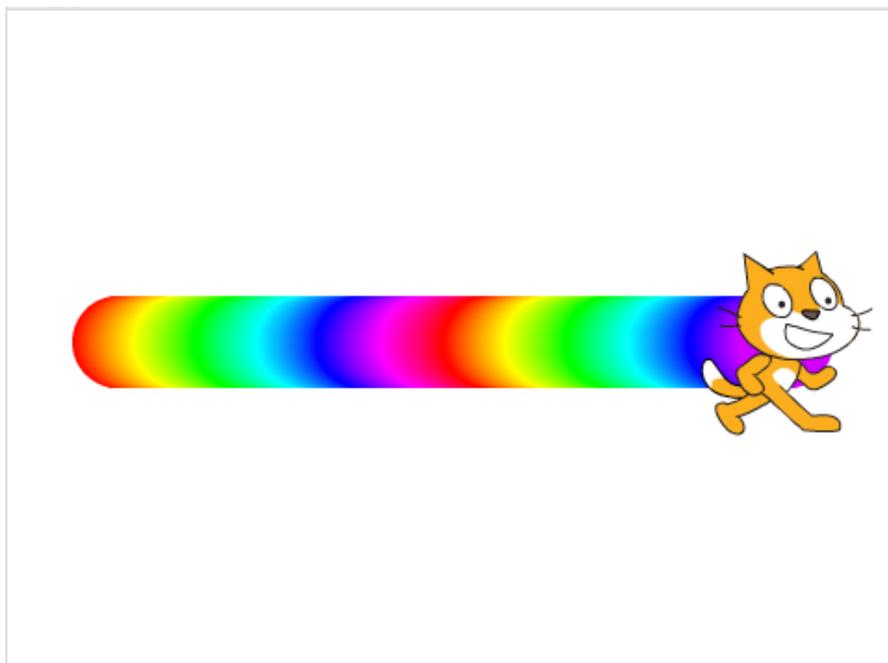


## 発展チャレンジ

### サンプルプログラム



下図はサンプルプログラムを実行したところです。虹色の線が描かれます。



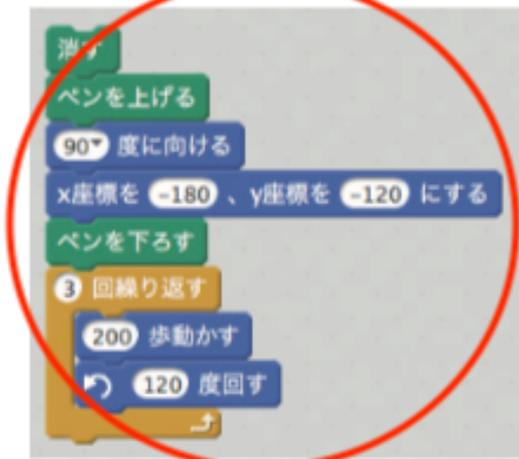
## チャレンジ 07

コードペディア (解答例)

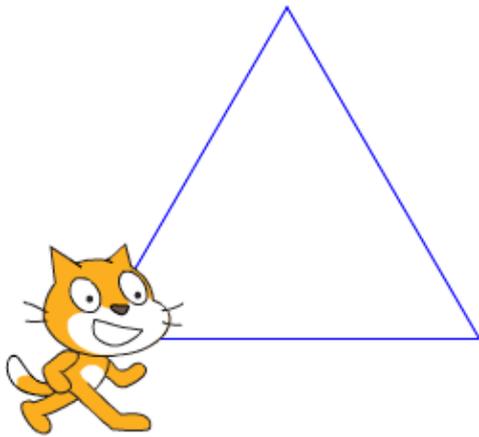


ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



上のサンプルプログラムを実行すると下図のような三角形を描きます。



## 発展チャレンジ

### サンプルプログラム



## チャレンジ 08

コードペディア (解答例)



ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



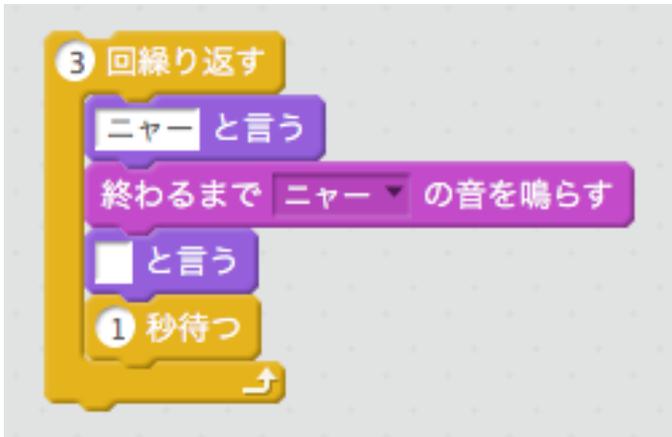
## 発展チャレンジ

### サンプルプログラム



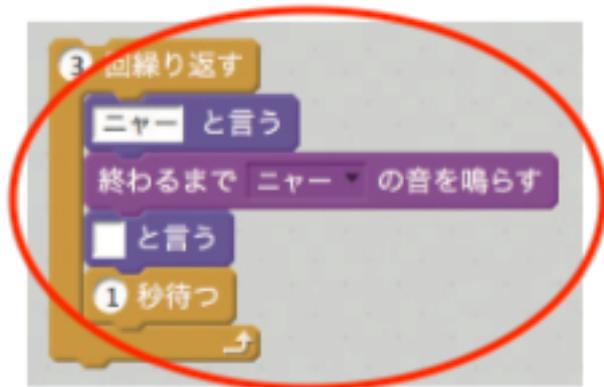
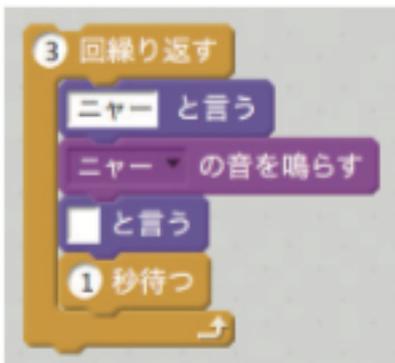
## チャレンジ 09

コードペディア (解答例)



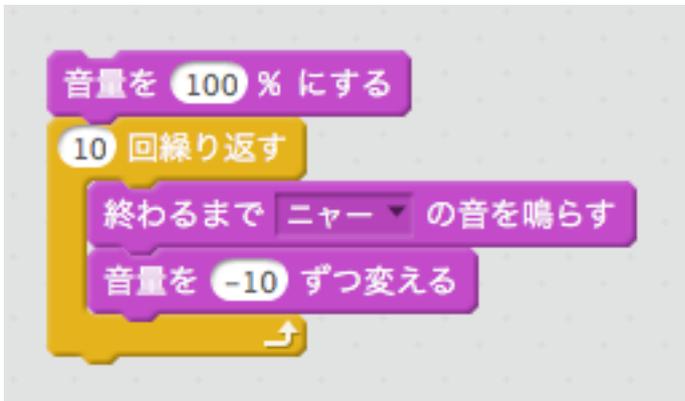
ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



## 発展チャレンジ

### サンプルプログラム



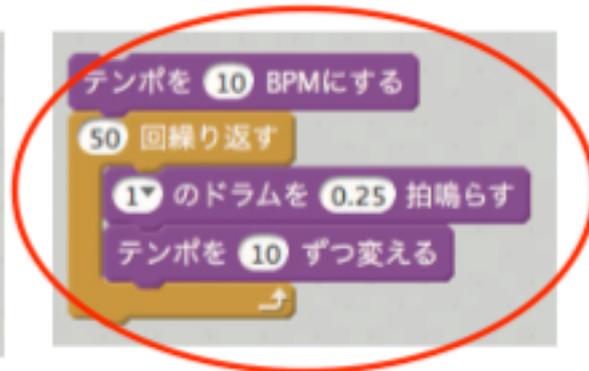
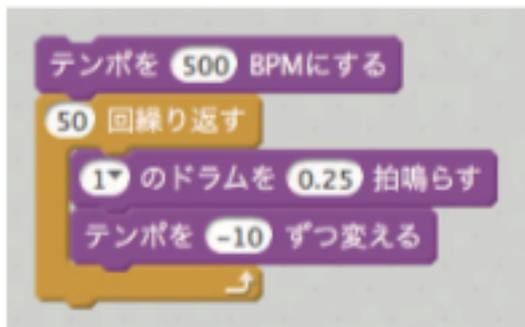
## チャレンジ 10

コードペディア (解答例)



ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



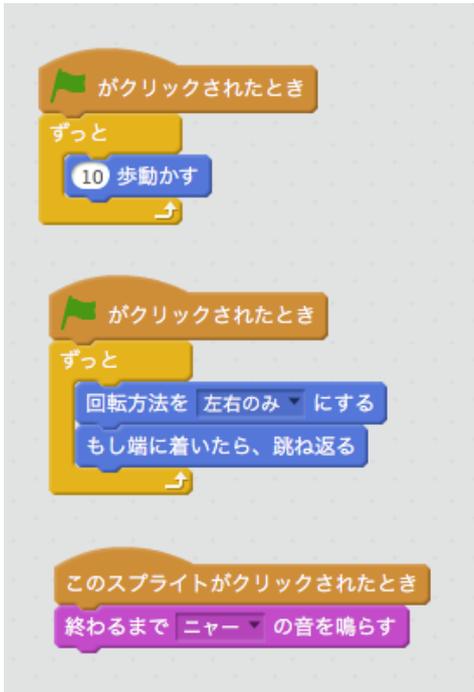
## 発展チャレンジ

### サンプルプログラム



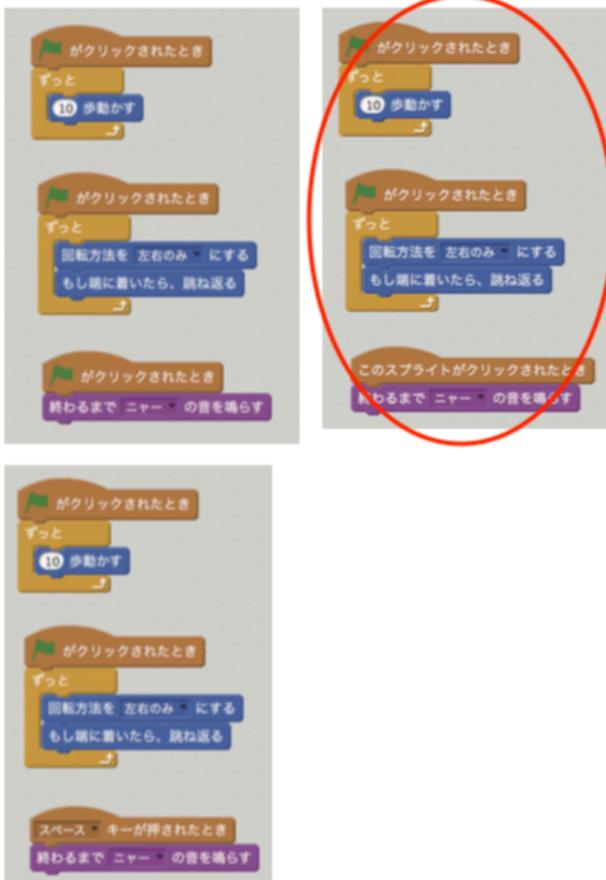
# チャレンジ 11

コードペディア (解答例)



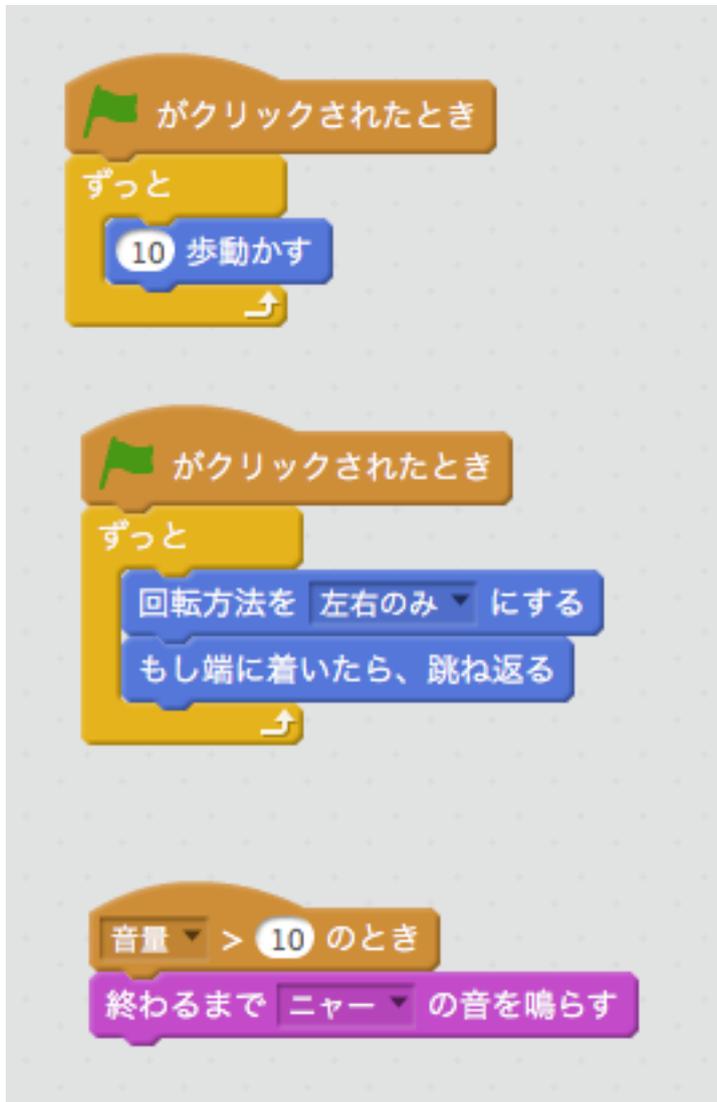
ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



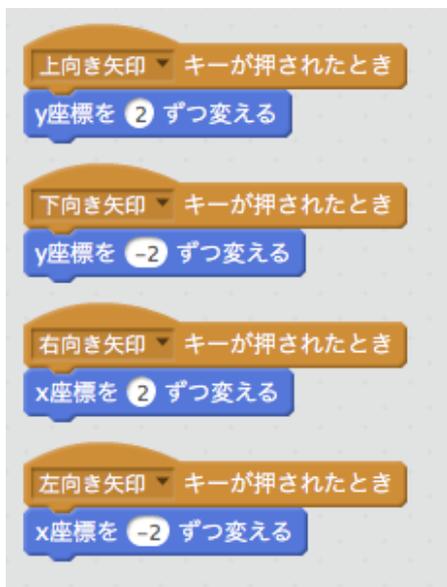
## 発展チャレンジ

### サンプルプログラム



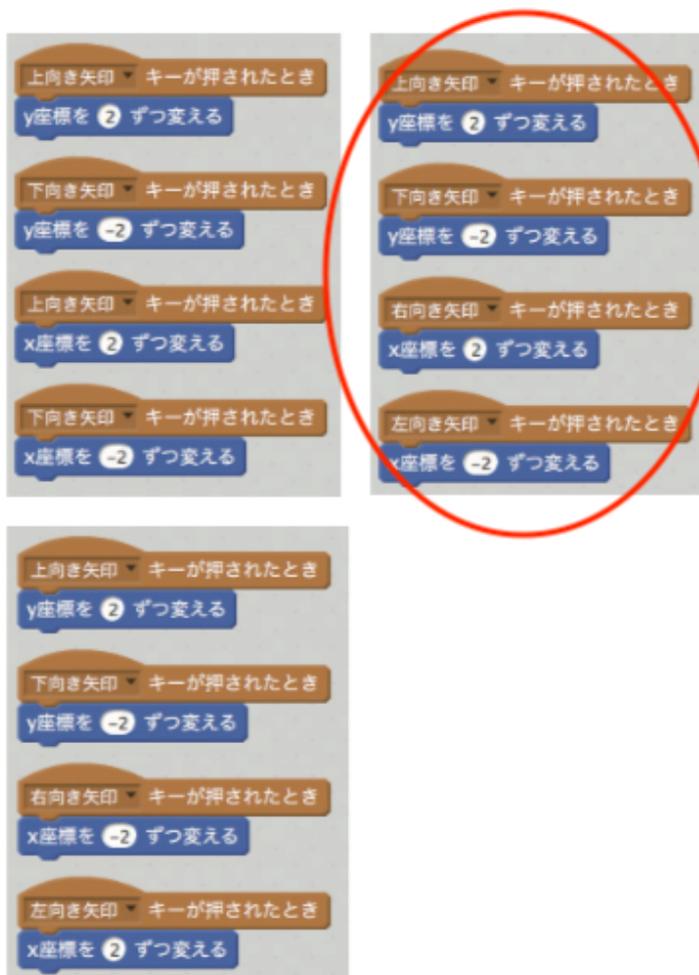
## チャレンジ 12

コードペディア (解答例)



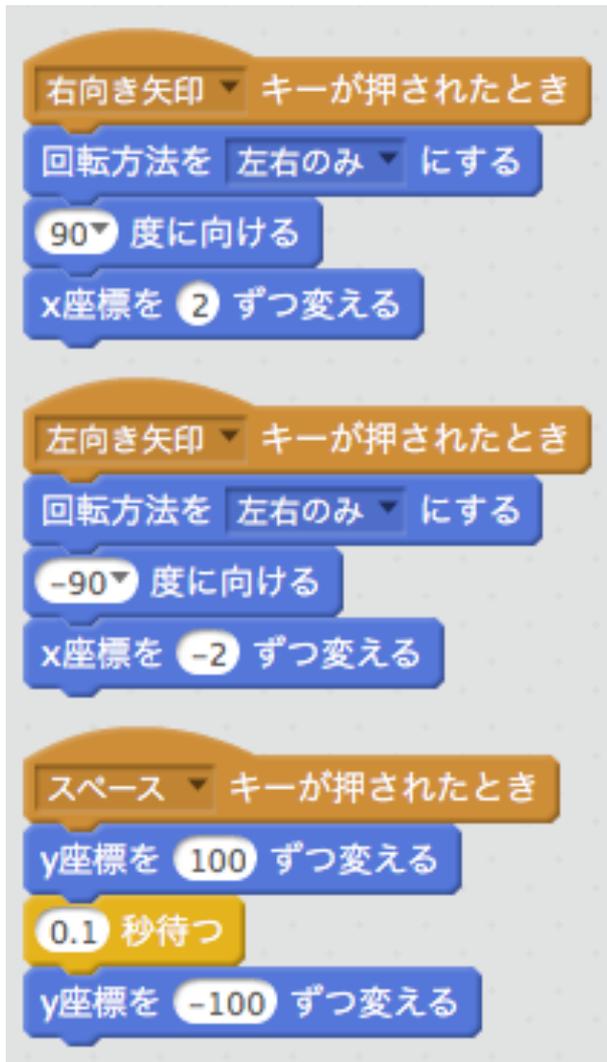
ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



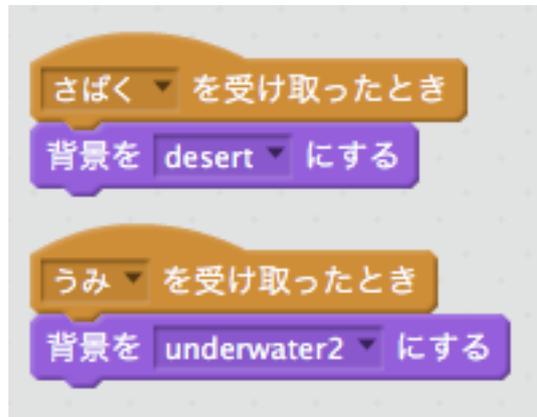
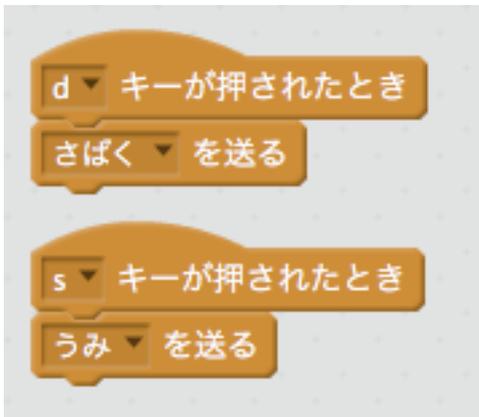
## 発展チャレンジ

### サンプルプログラム



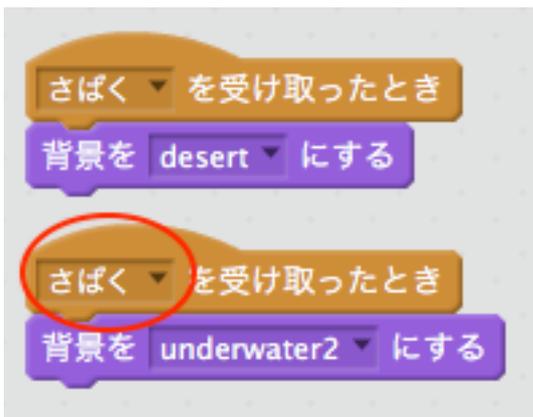
## チャレンジ 13

コードペディア (解答例)



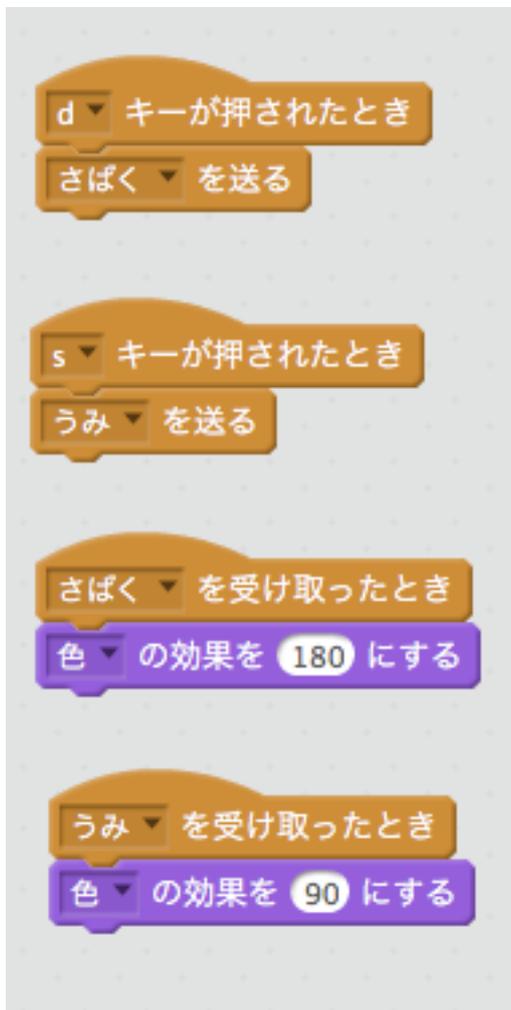
ヒント

下図で赤丸で囲まれた「さばく」を「うみ」に変更します。



## 発展チャレンジ

### サンプルプログラム



## チャレンジ 14

コードペディア (解答例)



ヒント

下図で修正しなくてはならないところは第1に下図の赤矢印が示す2箇所にある「y座標」を「x座標」にすること。第2に  $x > 100$  となった時に元の位置に戻す命令を追加することです。以上を修正すると上のコードペディア (解答例) のようになります。



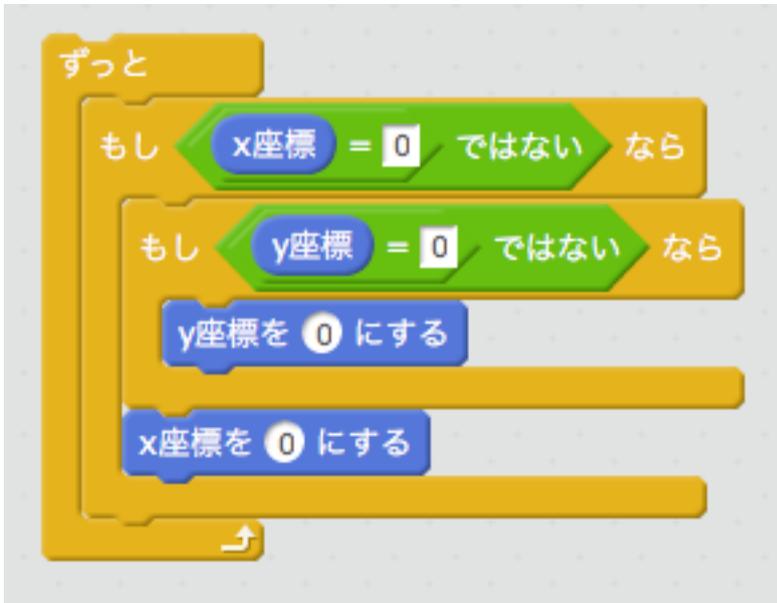
## 発展チャレンジ

### サンプルプログラム



## チャレンジ 15

コードペディア (解答例)



ヒント

ヒントに示されたプログラムのままだと、画面左にネコを置いた時には中央位置に戻りますが画面右側に置いた時には中央位置には戻りません。正しく動くようにするには上図のコードペディア (解答例) のようにします。

## 発展チャレンジ

### サンプルプログラム



## チャレンジ 16

コードペディア（解答例）

<訂正>

1)

誤：「スプライト2」は、スクラッチを起動した時にネコのスプライトについている名前です。

正：「スプライト1」は、スクラッチを起動した時にネコのスプライトについている名前です。

2)

誤：このプログラムのままだともともとのネコ（実体）をクリックした時だけ新しいクローンができます。クローンをクリックしても新しいクローンはできません。

正：このプログラムを使うと指示通りの動きとなります。

説明：

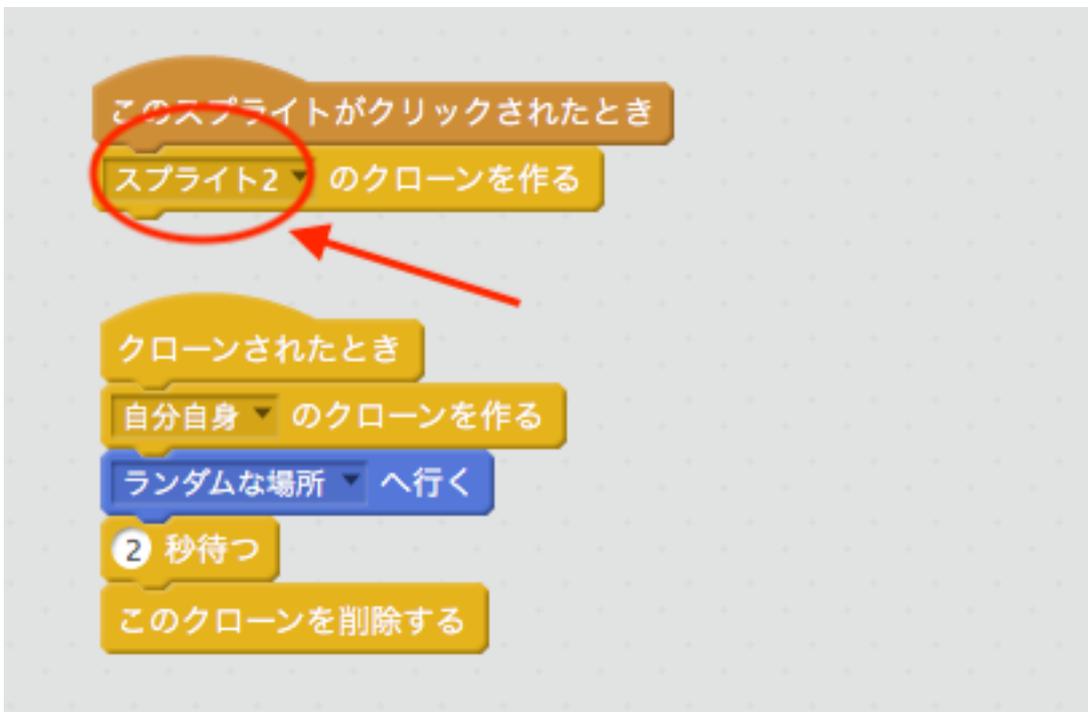
P.46で掲載したプログラムは正しい動きをするプログラムです。

「このプログラムのままだともともとのネコ（実体）をクリックした時だけ新しいクローンができます。クローンをクリックしても新しいクローンはできません。」のようになるのは下図のようなプログラムの場合です。



## 発展コードペディア

説明：P.47上側にあるプログラムで下図の赤矢印の先にある「スプライト2」という名前はスクラッチ直後であれば「スクラッチ1」となります。



ネコのスプライトの名前が今どうなっているかを確認するには次のようにします。

1) 下図の赤矢印の先にある「i」をクリックします。



2) 下図のようにスプライトの詳細情報が表示されます。赤丸で囲ったところがスプライトの名前になります。



**チャレンジ 17**

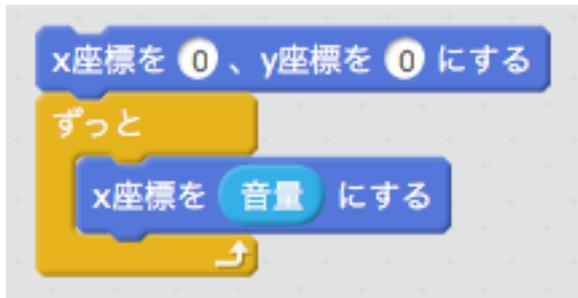
**チャレンジ 18**

全てのサンプルが紹介されています。サンプルプログラムを実際に試してみてください。

## チャレンジ19

コードペディア（解答例）

下図が P.54 のサンプルプログラムです。



このプログラムのままだと指示にある次の2点どおりになりません。

- 「音量」が 0 のとき：ネコは画面左端まで動く。
- 「音量」が 100 のとき：ネコは画面右端まで動く。

実際に上のサンプルプログラムを動かして試してみてください。ネコは画面の中央から右側に少し動くだけです。

下図は P.54 のサンプルプログラムを正しく修正したものです。



ヒント

P.55上側にあるプログラムを正しく修正したものが上図のプログラムとなります。実際に動かして違いを見てください。

## 発展チャレンジ

サンプルプログラムが P.56 に掲載してあります。試してみてください。

## チャレンジ20

全てのサンプルが紹介されています。サンプルプログラムを実際に試してみてください。

## チャレンジ21

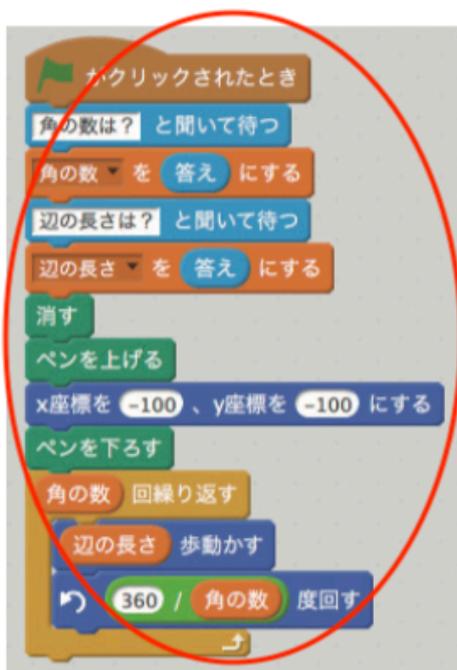
コードペディア（解答例）



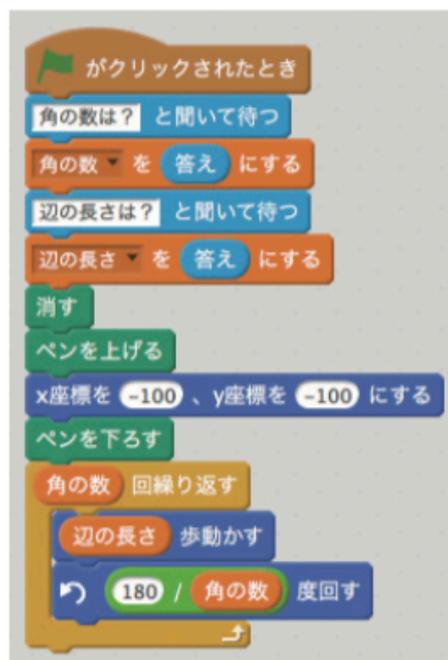
```
Scratch code for Challenge 21:
- When clicked:
  - Ask 'Number of sides?' and wait
  - Set 'Number of sides' to answer
  - Ask 'Length of side?' and wait
  - Set 'Length of side' to answer
  - Clear
  - Lift pen
  - Set x coordinate to -100, y coordinate to -100
  - Lower pen
  - Loop 'Number of sides' times:
    - Move 'Length of side' steps
    - Turn 360 / 'Number of sides' degrees
```

ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。



```
Scratch code with a red circle highlighting the loop:
- When clicked:
  - Ask 'Number of sides?' and wait
  - Set 'Number of sides' to answer
  - Ask 'Length of side?' and wait
  - Set 'Length of side' to answer
  - Clear
  - Lift pen
  - Set x coordinate to -100, y coordinate to -100
  - Lower pen
  - Loop 'Number of sides' times:
    - Move 'Length of side' steps
    - Turn 360 / 'Number of sides' degrees
```



```
Scratch code with a different loop value:
- When clicked:
  - Ask 'Number of sides?' and wait
  - Set 'Number of sides' to answer
  - Ask 'Length of side?' and wait
  - Set 'Length of side' to answer
  - Clear
  - Lift pen
  - Set x coordinate to -100, y coordinate to -100
  - Lower pen
  - Loop 'Number of sides' times:
    - Move 'Length of side' steps
    - Turn 180 / 'Number of sides' degrees
```

## 発展チャレンジ

サンプルプログラムが掲載されています。自分でも動かして試してみましょう。

## チャレンジ22

コードペディア（解答例）

```
消す
ペンを上げる
x座標を 0、y座標を -50 にする
0° 度に向ける
ペンを下ろす
辺の長さ を 15 にする
角度 を 7 にする
60 回繰り返す
  3 回繰り返す
    辺の長さ 歩動かす
    120 度回す
  辺の長さ を 3 ずつ変える
  角度 度回す
```

ヒント

ヒントのプログラムでは下図で赤丸で囲ったプログラムだと指示通りの動きとなります。

```
消す
ペンを上げる
x座標を 0、y座標を -50 にする
0° 度に向ける
ペンを下ろす
辺の長さ を 15 にする
角度 を 7 にする
60 回繰り返す
  3 回繰り返す
    辺の長さ 歩動かす
    120 度回す
  辺の長さ を 3 ずつ変える
  角度 度回す
```

```
消す
ペンを上げる
x座標を 0、y座標を -50 にする
0° 度に向ける
ペンを下ろす
辺の長さ を 15 にする
角度 を 7 にする
60 回繰り返す
  4 回繰り返す
    辺の長さ 歩動かす
    90 度回す
  辺の長さ を 3 ずつ変える
  角度 度回す
```

## 発展チャレンジ

P.66 にあるサンプルプログラムの空いたところを下図のようにすると、P.65の下にある図と同じものが描けます。



**チャレンジ23**

**チャレンジ24**

**チャレンジ25**

全てのサンプルが紹介されています。サンプルプログラムを実際に試してみてください。

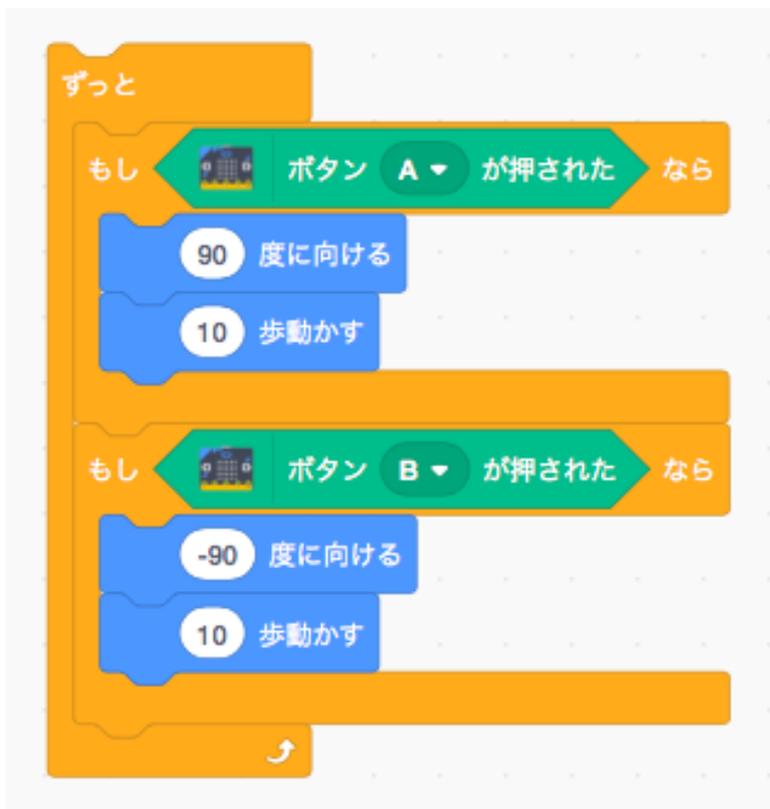
## チャレンジ26

コードペディア（解答例）



説明：本書執筆時にmicro:bit 関係の命令が英語のままであったものが 2018年10月現在のベータ版では日本語に翻訳されています。上図の命令「ボタン ( ) が押された」もその例です。

ヒント（解答例）



## 発展チャレンジ

サンプルプログラムが掲載されています。自分でも試してみましょう。

## チャレンジ27

コードペディア（解答例）



説明：本書執筆時にmicro:bit 関係の命令で、2018年10月現在のベータ版では命令の名前が変更されているものがあります。上図の命令「display text ( )」もその例です。2019年1月に予定されている正式版リリースまでは命令の名前などは変更される可能性があります。

ヒント

ヒントのサンプルプログラムが掲載されています。自分でも試してみましょう。

## 発展チャレンジ

サンプルプログラムが掲載されています。自分でも試してみましょう。

## チャレンジ28

コードペディア（解答例）



説明：本書執筆時にmicro:bit 関係の命令が英語のままであったものが 2018年10月現在のベータ版では日本語に翻訳されています。上図の命令「 ( ) を表示する」、「画面を消す」もその例です。

ヒント（解答例）



## 発展チャレンジ

サンプルプログラムが掲載されています。自分でも試してみましょう。

## チャレンジ29

コードペディア

サンプルプログラムが掲載されています。自分でも試してみましょう。

発展チャレンジ

自分のアイデアで様々な改良してみてください。

## チャレンジ30

コードペディア（解答例）



説明：本文では省略されていますが、準備として変数「テンポ」を自分で作っておく必要があります。

説明：本書執筆時にmicro:bit 関係の命令が英語のままであったものが 2018年10月現在のベータ版では日本語に翻訳されています。上図の命令「ボタン（ ）が押された」もその例です。

発展チャレンジ

サンプルプログラムが掲載されています。自分でも試してみましょう。

# スクラッチ ドリルブック

作って学ぶ実践プログラミング練習帳

石原 正雄 ● 著

HAUNT WORKS STUDIO / 原本優莉 ● 協力



CUTY  
カットシステム