

おうちで作ろう パソコンゲーム！

お約束

パソコンで作業するときは30分ごとに
10分休憩しましょう！！



ロボットの基本要素



しじ
指示

あたま

コンピュータ

どうさ
動作

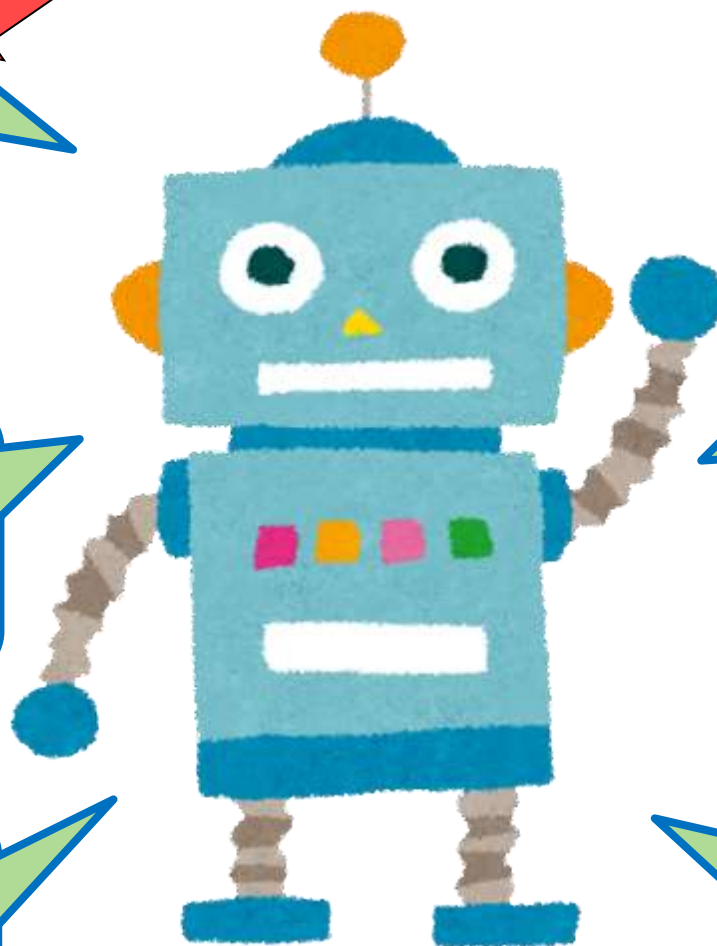
きんにく

アクチュエータ

こうぞう
構造

ほね

きこう
機構



ひとは…

ロボットは…

かんかく
感覚

め・みみなど

センサ

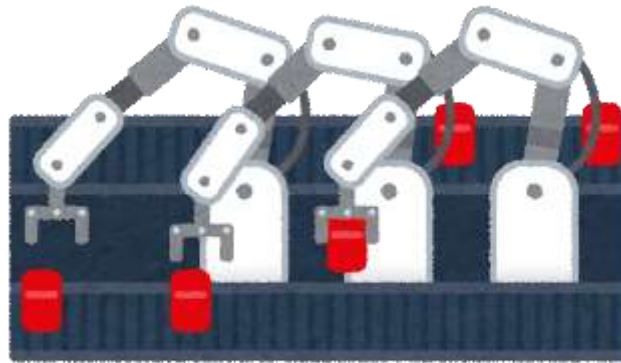
げん
エネルギー源

たべもの

バッテリーなど



現在、私たちの生活の中では、あらゆるものにコンピュータが搭載されています。では、コンピュータはどのように動くのでしょうか？





コンピュータを動かすには**プログラム**が必要

■ 「プログラム」とは？

» やってほしいことをさせるための**指示書**

■ 「プログラミング」とは？

» コンピュータがわかるように**指示書を書くこと**

» 手順を考えて指示をする必要があります

あなたが学校のテストを受ける時の手順を書いてみましょう

- 1.
- 2.
- 3.
- 4.
- ...



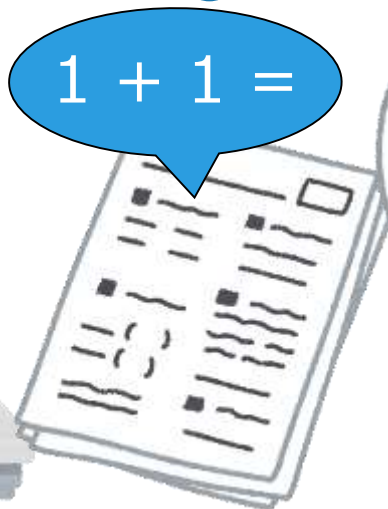
■ 例：テストを受けるときの手順

1. 問題を見る
2. 問題の内容を理解する
3. 問題の答えを考える
4. 考えたことを書く

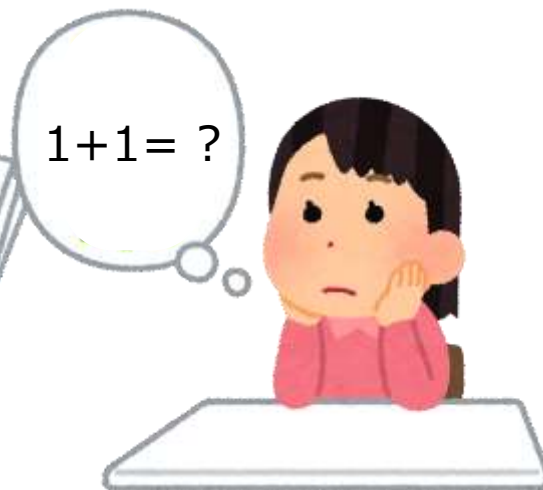
①



②



③



④





あなたがスタートから2m先のゴールまで

歩く手順を考えてみましょう！ ※ 1歩を25cmで考えてみましょう

①

⑨

②

⑩

③

⑪

④

⑫

⑤

⑬

⑥

⑭

⑦

⑮

⑧

⑯



回答例 1

- ① スタート
- ② 右足(または左足)を25cm前に出す
- ③ 左足(または右足)を50cm前に出す
- ④ 右足(または左足)を50cm前に出す
- ⑤ 左足(または右足)を50cm前に出す
- ⑥ 右足(または左足)を50cm前に出す
- ⑦ 左足(または右足)を25cm前に出す
- ⑧ ゴール

回答例 2

- ① スタート
- ② 右足(または左足)を25cm前に出す
- ③ ④と⑤を2回繰り返す
- ④ 左足(または右足)を50cm前に出す
- ⑤ 右足(または左足)を50cm前に出す
- ⑥ 左足(または右足)を25cm前に出す
- ⑦ ゴール

このようにプログラムにはいくつかの正解があります
思った通りに動けば、それが正解です！



いろいろな指示書の書き方（言語）があります

■ 機械語

(6502 ファミコンCPUが得意)

A9 0A 8D 00 50 ...

■ Python (パイソン)

(データ分析が得意)

```
import math
def bisection_method(a, b, eps):
    s = 0
    for i in range(1000):
        if(abs(a-b)<eps): break
        s = (a+b)/2.0
        if(f(s) * f(a)<0): b = s
        else: a = s
    return s
...
```

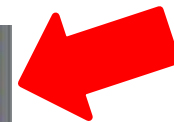
■ C言語 (スマホのアプリが得意)

```
#include<stdio.h>
int main(void){
    int a, b, tmp;
    printf("a = ");
    scanf("%d", &a);
    printf("b = ");
    scanf("%d", &b);
    ...
```

■ Scratch



今回使用する言語





AIとは



人工知能の種類

人間のように考える機械を作る
(強いAI)



人間が知能を使ってすることを機械にさせる
(弱いAI)



それに、関わる科学や**技術**の事



■ 自動運転 (Auto Pilot)



目的地まで運ぶ



安全に走る (物体認識など)

自動運転の自動車もプログラムで動いているよ!

身の回りにもたくさんのプログラムがかかっているよ 探してみてね



プログラミングで 大切なこと



コンピュータさんが
わかるように書く



思った通りに
動かないこともある



でも頑張って
原因を探す



- 今回は、簡単なゲーム作りを通してプログラミング(プログラム作り)を体験します



SCRATCH

<https://scratch.mit.edu/>

こんなゲームを作ります





- すでにあるブロックは消さない
- 資料と自分の画面を見比べる
- 動作確認しながら作成



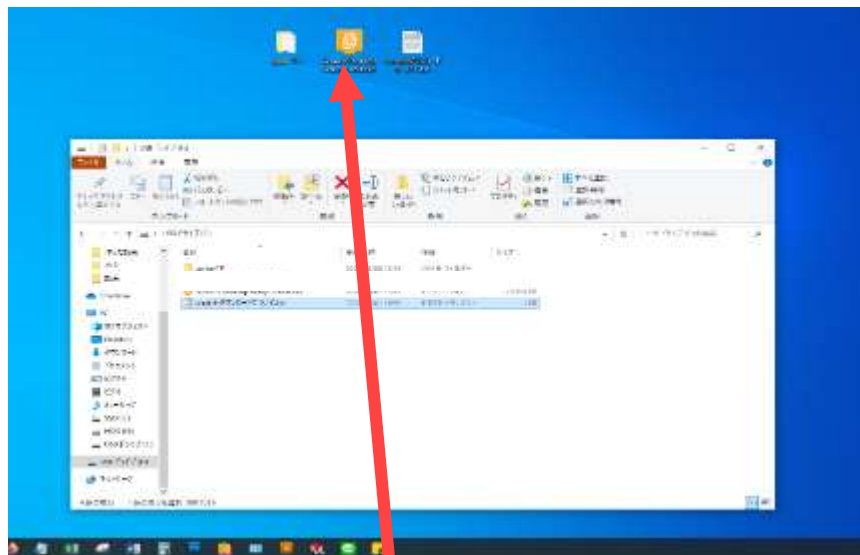
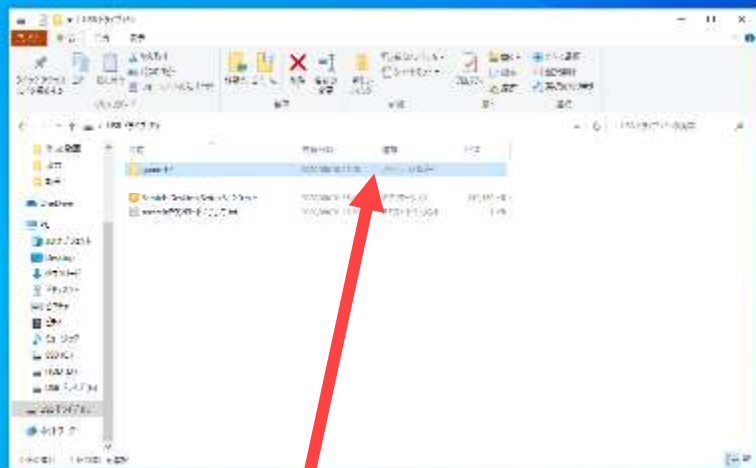
■使用するパソコンにUSBメモリ

に入っているファイルを全て

デスクトップにコピーして

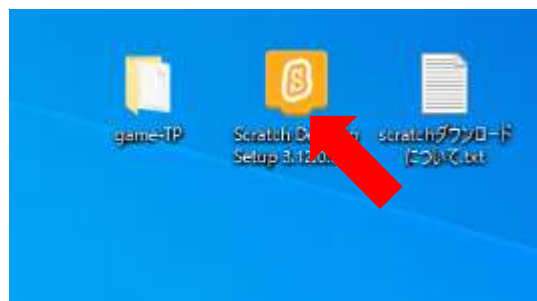
USBメモリを外す

※ もし間違っってパソコン上のファイルを消したり
壊してしまったらもう一度USBメモリからコピーしてください

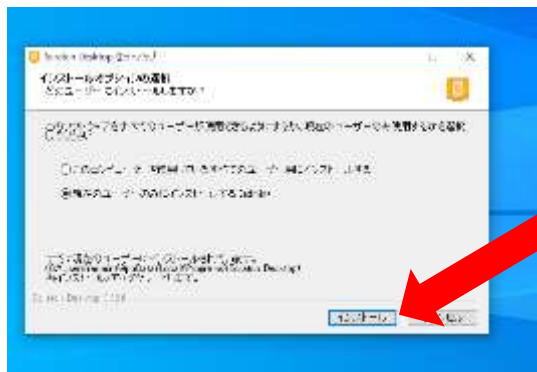


コピーしたいファイルの上で
マウスの左を押したまま

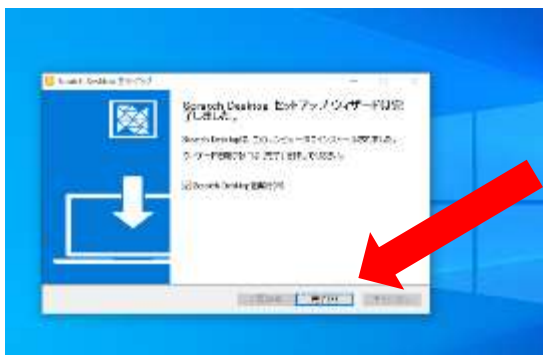
何もないところでマウスから
指をはなす



このファイルの上で
マウスの左を2回クリック



インストール をクリック



完了 をクリック

デスクトップに



できたかな？

Scratchを起動しよう！



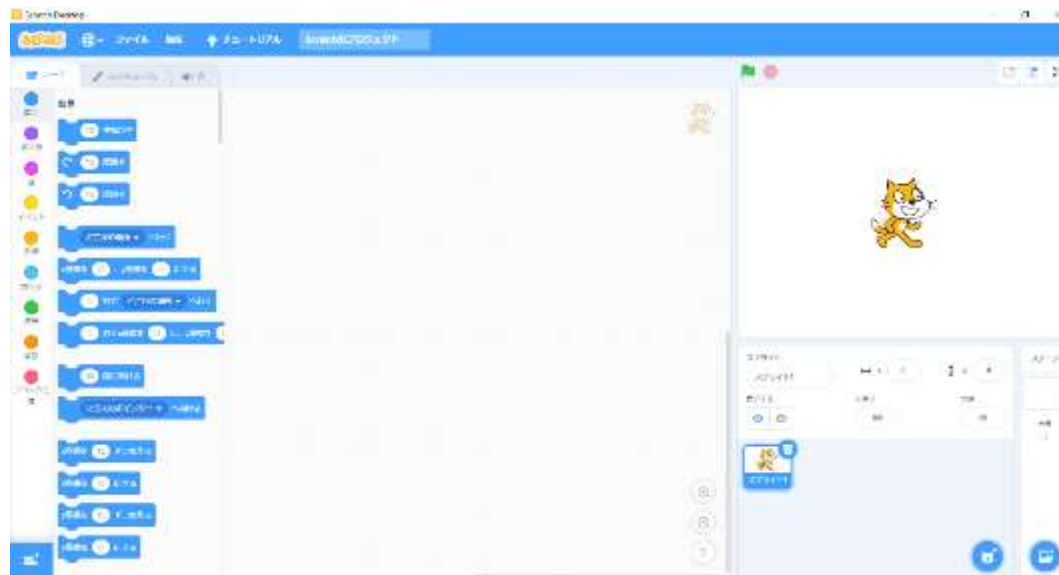
- Scratchを起動しましょう きどう



ダブルクリック

同じパソコンを
使用の場合
2回目以降は
ここから
スタート！

scratchの画面





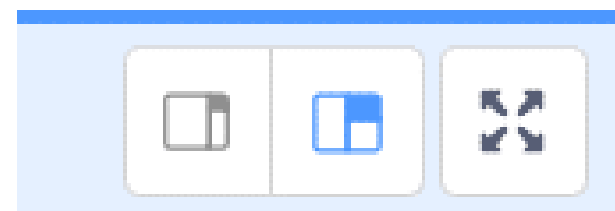
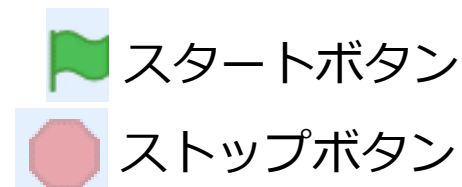
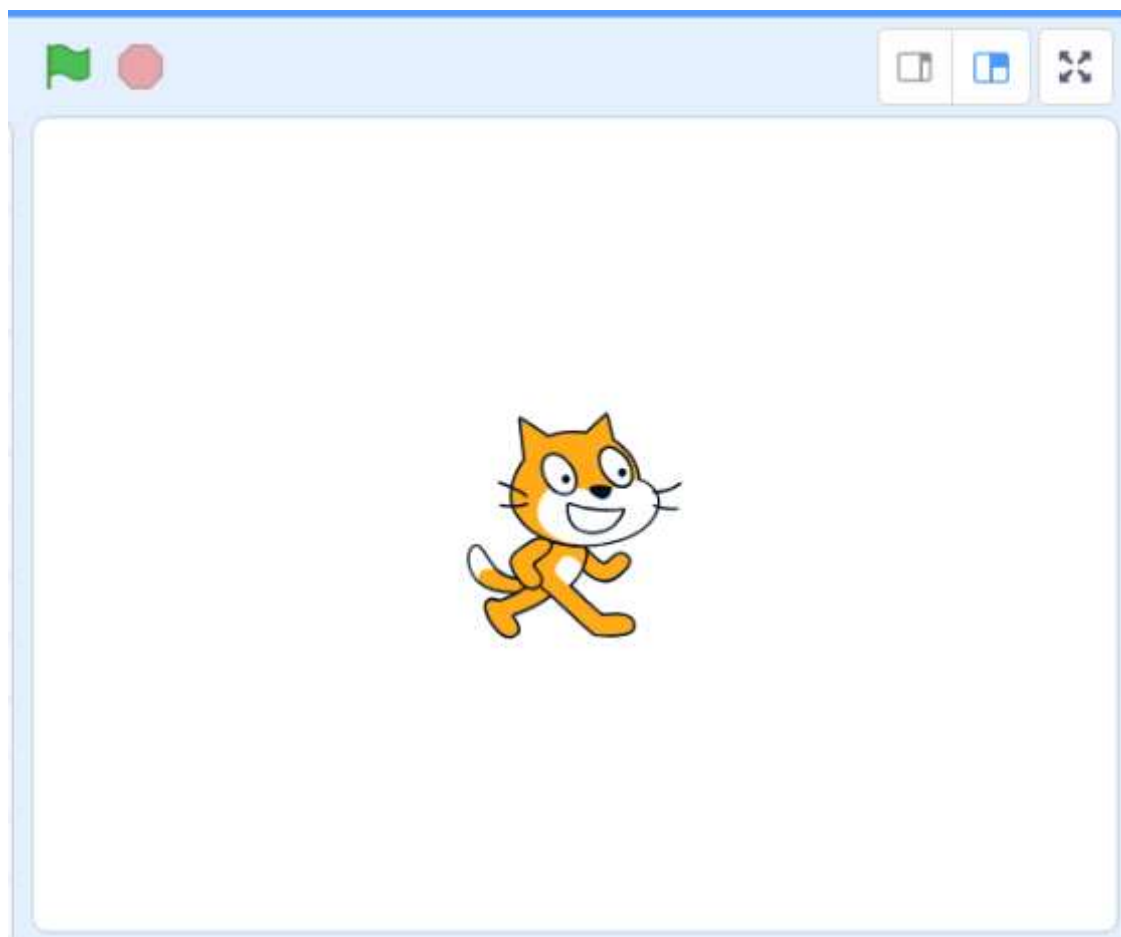
基本的な使い方①

The image shows the Scratch Desktop interface with several key areas highlighted and labeled:

- スタート** (Start) and **ストップ** (Stop) buttons are highlighted in a red box at the top right.
- The **プログラムの動く場所** (Program execution area) is the large white stage area on the right, containing the Scratch cat character.
- The **プログラムの置き場所** (Program placement area) is the central workspace where code blocks are placed.
- The **ブロックの種類を選ぶ場所** (Block selection area) is the left sidebar containing various colored block categories.
- The **ブロック置き場** (Block placement area) is the bottom section of the sidebar where specific code blocks are stored.
- The **スプライト(キャラクタ)を選ぶ場所** (Sprite selection area) is the bottom right panel where a character (sprite) is chosen for the program.



■ 実際にプログラムが動くのはここです



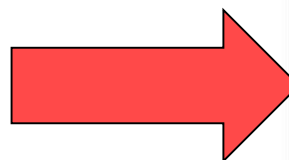
画面を大きくしたり
小さくしたりできます



基本的な使い方③

■ ブロックは種類があります

種類を選ぶ



バーを動かして探す



■ ブロックを左から選んでドラッグ&ドロップ

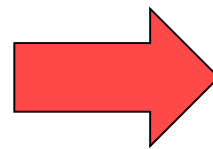
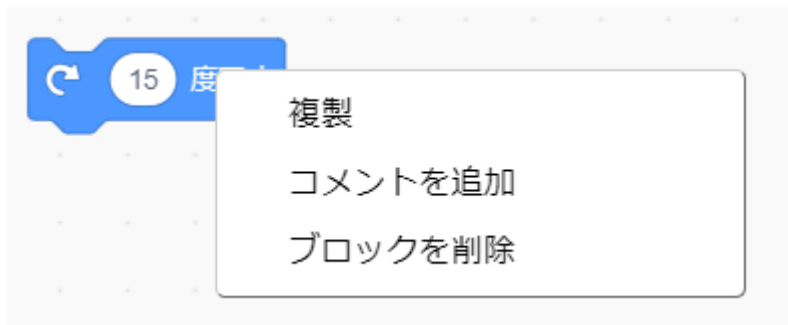




- ブロックを引っ張ると外れます



- コピーしたいときは、“複製”をクリック
- 使わないものは右クリックで“削除”できる





- ブロックは影がでたところにくっつく





- ブロックの空欄には、文字を入れるだけでなく
角が丸いブロックがはまる

例: こんにちは! のあと、マウスとの距離を数字で言います。
猫に近いほど小さい数字になります。



(ブロックはダブルクリックでそこだけ実行できる)



変数の作り方

■ 変数 へんすう

数字や文字を入れておける、入れ物です。

変数を作る

で好きな名前で作れます。

例：猫がクリックされると、クリックされた数を数えます



クリックされた数 8



右クリックで見た目を変えることができます。



掴んで動かすこともできます。





■ 保存は“ファイル”をクリック

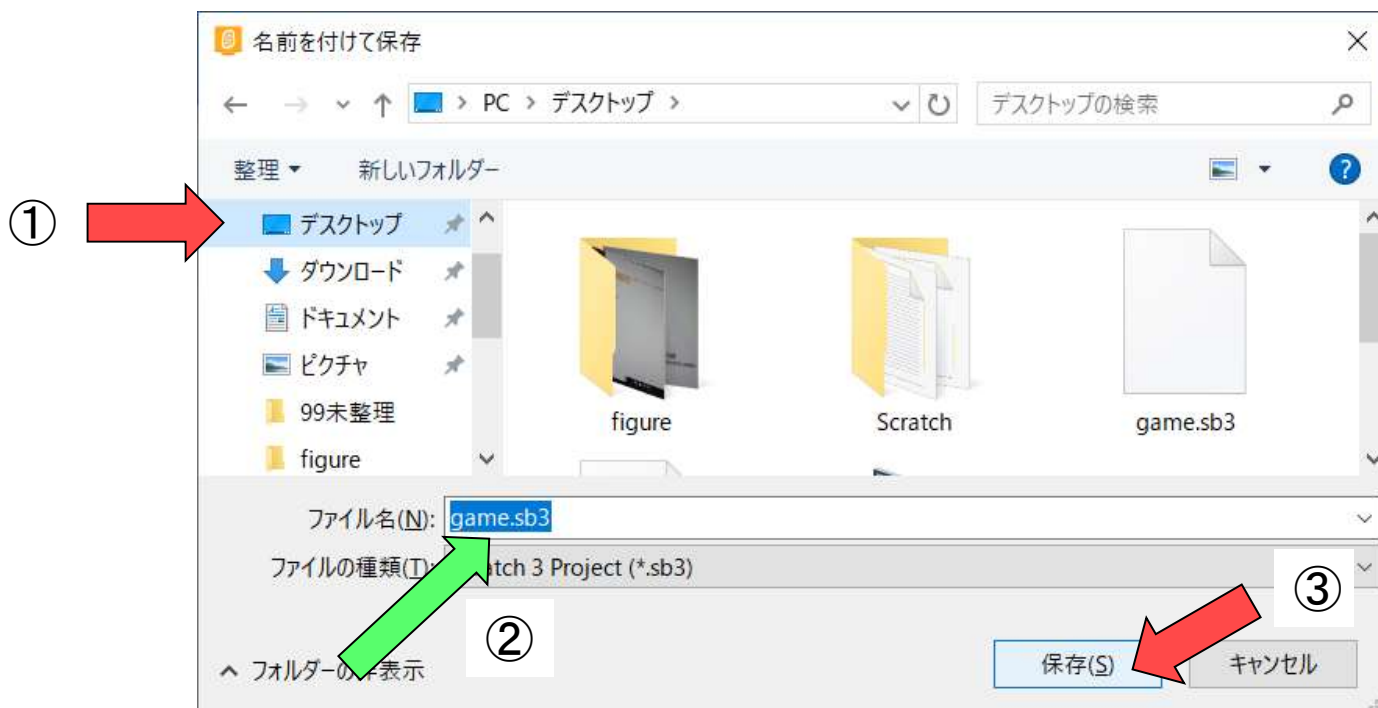


■ “コンピューターに保存する” をクリック



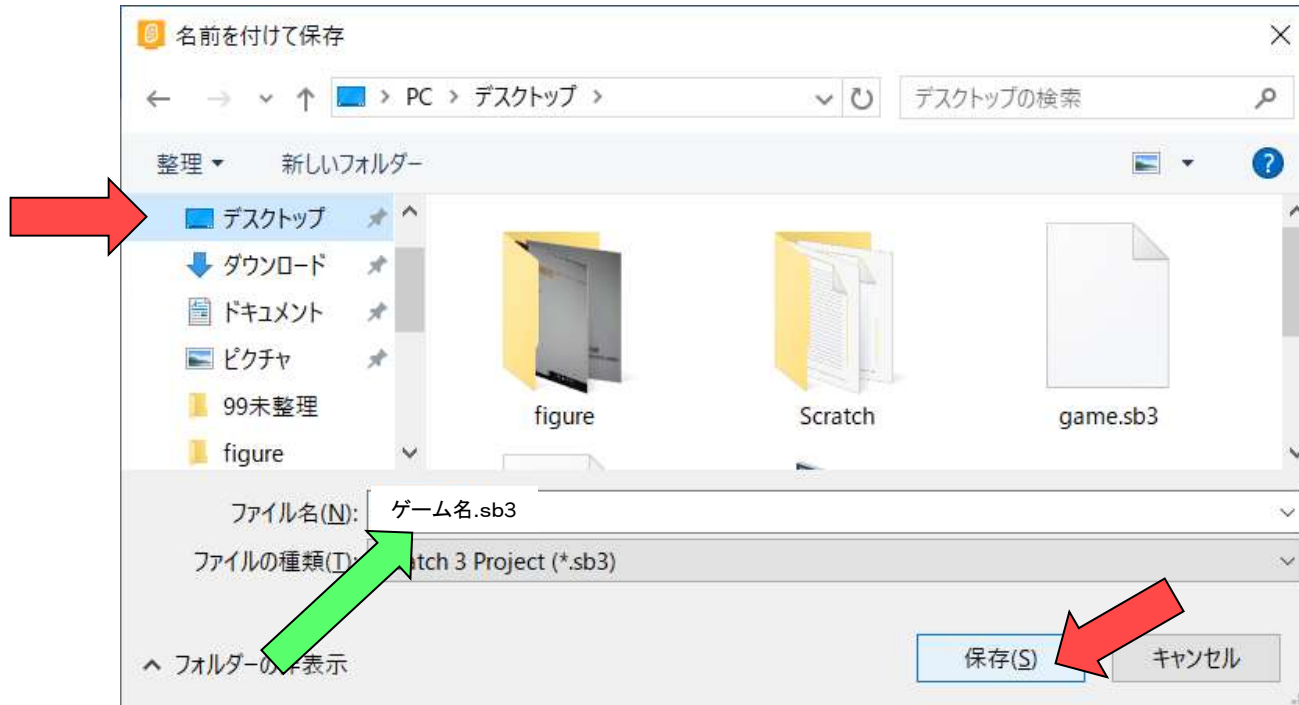


保存① 新しいゲームとして保存したいとき

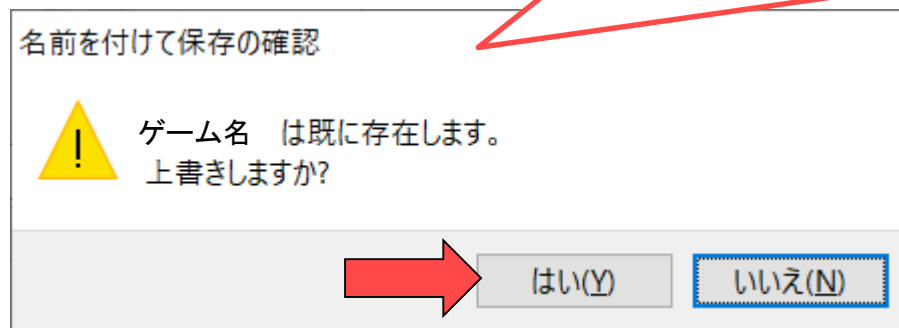


- ① 保存したい場所を選ぶ（今回はデスクトップ）
- ② 英数字で新しい名前をつける
- ③ 保存 をクリック

保存② ゲームを修正した時



こんな表示が出ても、“はい”をクリック





実際にゲームを
作ってみよう！

ゲームのテンプレートをひらく



① “ファイル”を選ぶ

② “コンピュータから読み込み”を選ぶ



③ “デスクトップ”を選ぶ

④ “game-TP”を開く



⑤ “template”を開く



- まずはサンプルを完成させましょう！
- あらかじめ、ステージは作ってあります。
- しかし、ムロぴよんを動かす部分がありません

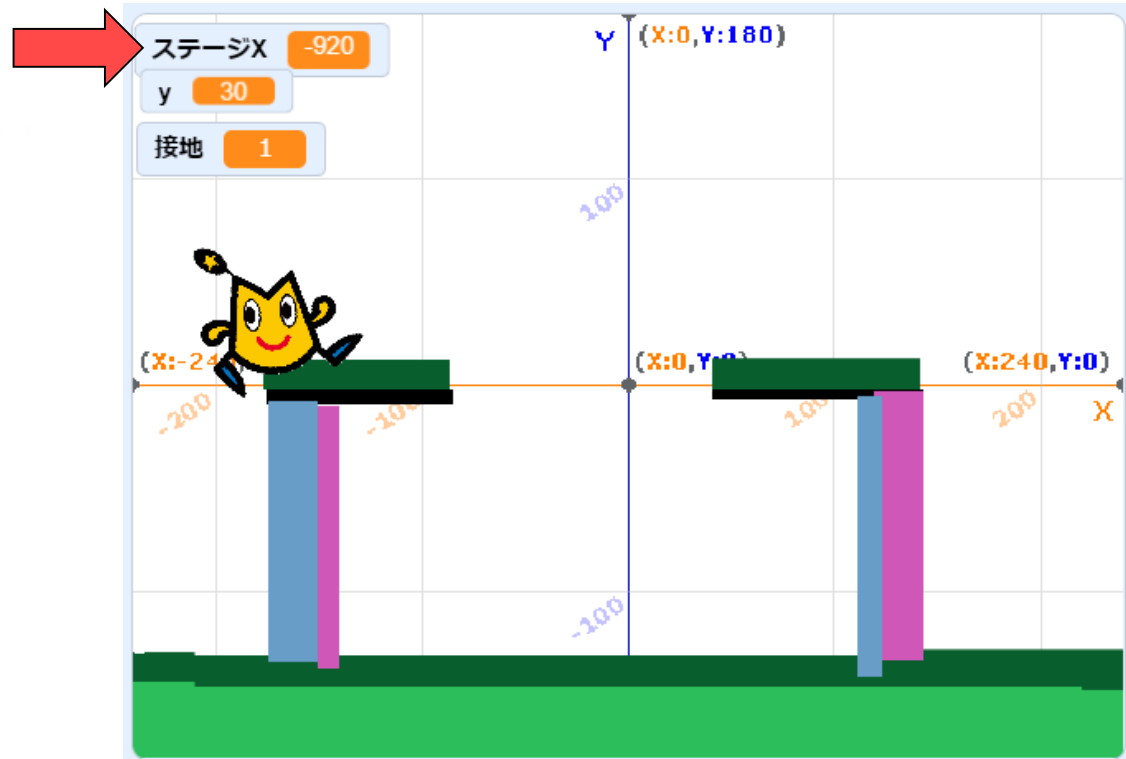


ムロぴよんが動く仕組み



そもそもムロぴよんはどう動いているのか？
■ でもムロぴよんは画面の中しか動けません
ステージが動けばいいんです。

どれだけ
ステージが動いたか





ステージ1を選択し、
右のプログラム
追加する





- ムロぴよんが浮いていてはゲームになりません。床に足をつけたいですね。
- 資料の次のページにあるブロックを新しく追加してください。(今までののは消さずに)
- 色は、スポイトで取ります。

スポイトの使い方



ほしい色のところをクリック





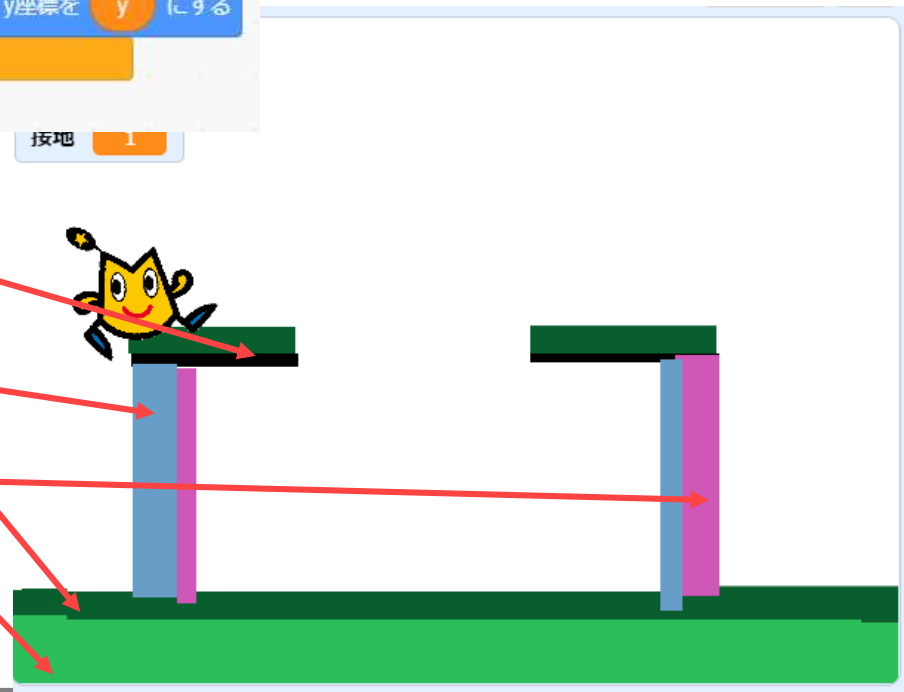
ムロぴよんを選択して、
プログラムを追加。





```
が押されたとき
ずっと
もし 緑色に触れた なら
  接地 を 1 にする
でなければ
  接地 を 0 にする
もし 青色に触れた なら
  y を 10 ずつ変える
  x座標を -166 、y座標を y にする
もし 黒色に触れた なら
  y を -10 ずつ変える
  x座標を -166 、y座標を y にする
もし 水色に触れた なら
  ステージX を 10 ずつ変える
  x座標を -166 、y座標を y にする
もし 紫色に触れた なら
  ステージX を -10 ずつ変える
  x座標を -166 、y座標を y にする
```

```
が押されたとき
  最前面へ移動する
  x座標を -166 、y座標を 100 にする
  y を 100 にする
  ずっと
    もし 接地 = 0 なら
      y を -10 ずつ変える
      x座標を -166 、y座標を y にする
```





- カベと重力が仕事するようになったので、今度はゴールできなくなっていました。
- ジャンプさせてみましょう！
- **ブロックを新しく追加**してください。

**ムロピョンを選択して、
プログラムを追加。**





ゴールまで行けたらサンプル完成です。





ゲームをもっと
発展させよう！



その①

てき しゅつげん
敵キャラの出現



■ 敵を追加する



ここをクリックして、新しいスプライトを追加する。

好きなスプライトを選択する



今回は「bat」を選択します



大きさを調整する今回は、「50」にする

スプライト

Bat

↔ x 265

↕ y 0

表示する

大きさ 50

向き 90

muropyon

スプライト1

スプライト2

スプライト3

スプライト4

Cat

スプライト5

Bat

A screenshot of the Scratch software's sprite editor. The 'Bat' sprite is selected and highlighted in blue. The '大きさ' (Size) property is set to 50 and is circled in red. Other properties shown include 'x' (265), 'y' (0), and '向き' (Direction) (90). The sprite palette below shows various options including 'muropyon', 'Cat', and several unnamed 'スプライト' (Sprites).



新しく追加したスプライトに右のプログラムを追加する。

```
が押されたとき
y座標を -50 にする
ずっと
  20 回繰り返す
  y座標を 10 ずつ変える
  1 秒待つ
  20 回繰り返す
  y座標を -10 ずつ変える
  0.5 秒待つ
```

```
が押されたとき
隠す
x座標を ステージX + 1100 にする
ずっと
  x座標を ステージX + 1100 にする
  もし ステージX + 1100 < 300 なら
    表示する
  もし ステージX + 1100 < -300 なら
    隠す
  もし muropyon に触れた なら
    ゲームオーバー を送る
```

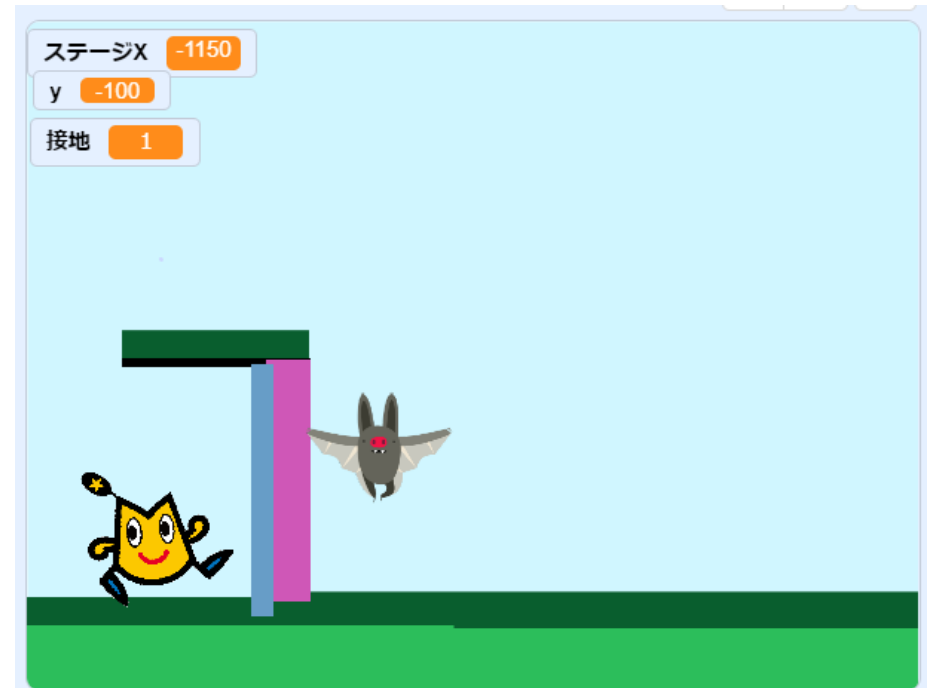
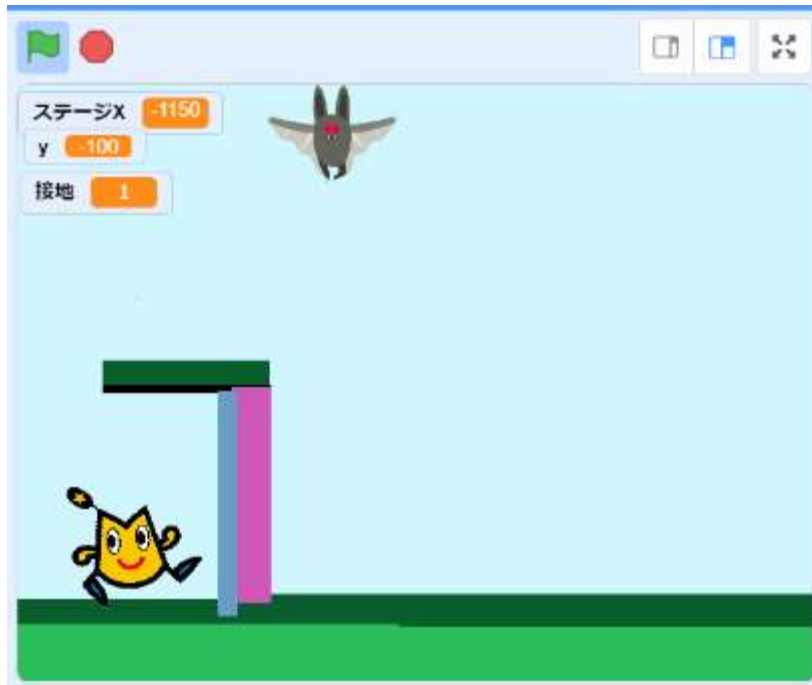



ゲームオーバーの sprites を選
択して、プログラムを追加する。





このように表示されたら完成です





その②

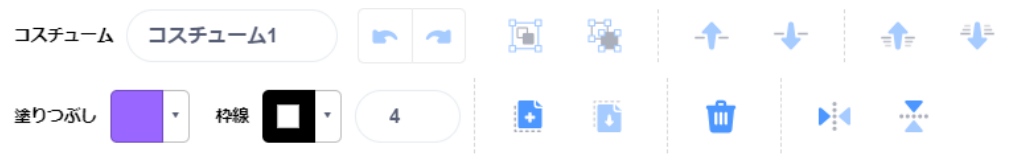
むずか

難しいステージ



■ ステージ拡張しよう。

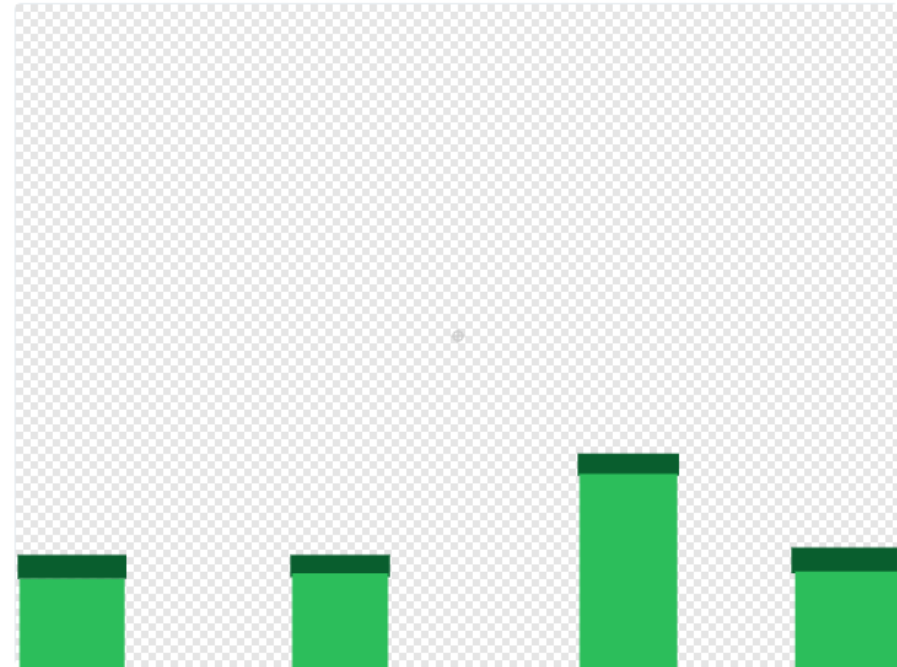
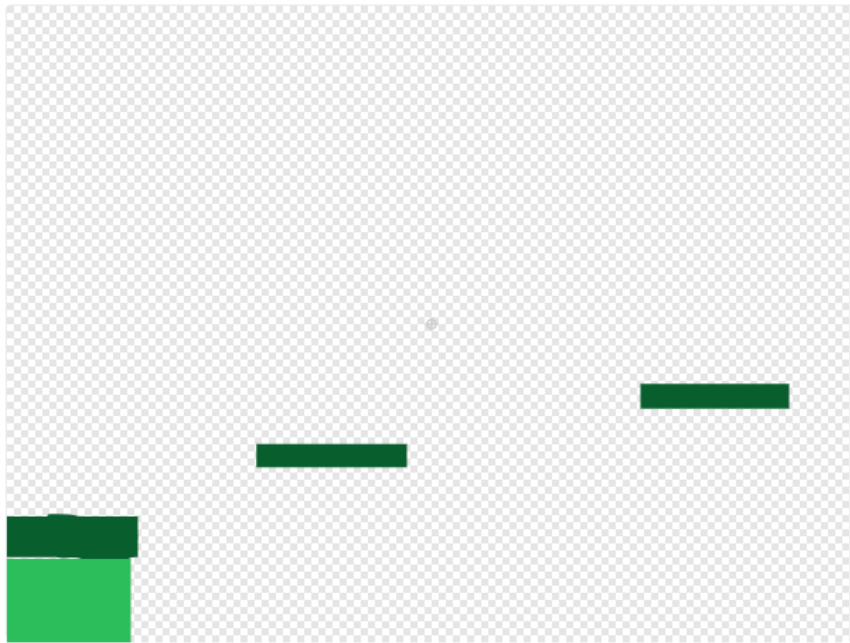
描くを選択して、



ビットマップに変換する。



ステージの色が他のステージと同じ色になるように
このように好きな形のステージを作ってください。





新しく追加したステージにこのプログラムを追加してください。

```
が押されたとき
  最背面へ移動する
  ずっとループ
    x座標を ステージX + 1700、y座標を -10 にする
    もし ステージX + 1700 < -480 なら
      隠す
    もし -480 < ステージX + 1700 なら
      表示する
```



猫の位置を変更します。



これらの数値を変更して猫の位置を調整してみてください。

