

第3章 チャレンジ ステップアップ 解答と解説

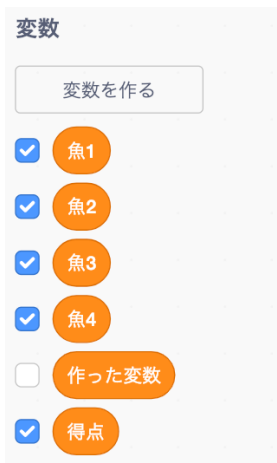
チャレンジ 3.1.c1



さめのスプライトの「得点を○ずつ変える」の数を2にします

ステップアップ 3.1.s1

クリックされた回数を記録する変数をそれぞれの魚に対して作ります。



1. 変数を作ります。



2. 初期化を忘れないようにしましょう。ステージのスク립トを修正します。

3. それぞれの魚のスプライトで、「得点を1ずつ変える」のブロックの下に「魚○を1ずつ変える」のブロックを追加します。



動作がおかしいときは、「1ずつ変える」と「1にする」を間違っていないかどうかをチェックしましょう。

チャレンジ3.3.c1



答えが間違ったとき、なので「もし（答え＝正解）なら」の、「でなければ」のところに、「得点を-2ずつ変える」のブロックを追加します。

チャレンジ3.3.c2



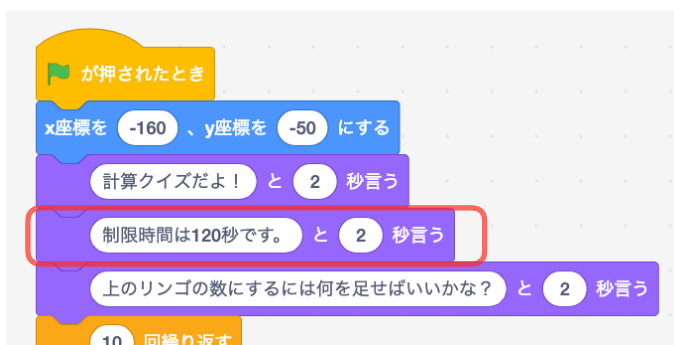
1. プログラムがスタートしてから○秒後に処理をする方法として、「○秒待つ」というブロックを使います。

2. ステージのスクリプトに「120秒待つ」ブロックを追加します。その次に「おわり」のメッセージをネコのスプライトに送るように、「おわりを送る」のブロックを追加します。ネコのスプライトには、「おわり」を受け取った時の処理を追加します。

ステージ

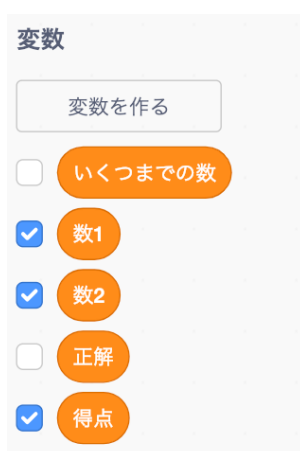


ネコのスプライト



3. 問題を出す前に、ネコに「制限時間は120秒です」というせりふを言わせると親切ですね。

ステップアップ 3.3.s1



1. 「いくつまでの数」という変数を作ります。チェックを外して、表示しないようにします。



いくつまでの数にしますか?と尋ねて、ユーザーに入力をしてもらいます。

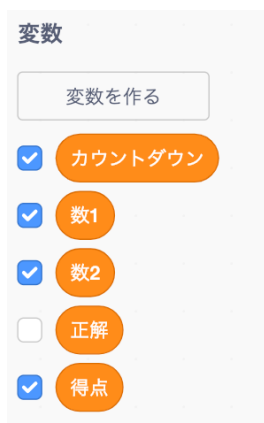
ユーザーの入力を変数「いくつまでの数」に入れます。

「1~いくつまでの数」までの乱数を作って「数1」に入れます。

ステップアップ 3.3.s2

制限時間である120秒以内に10問答えなければ時間切れです。正解でも不正解でも、とにかく10問答えなければなりません。

Scratchプログラミングで時間を計る方法として、前述した「〇〇秒待つ」ブロックのほかに「タイマー」ブロックを使う方法があります。



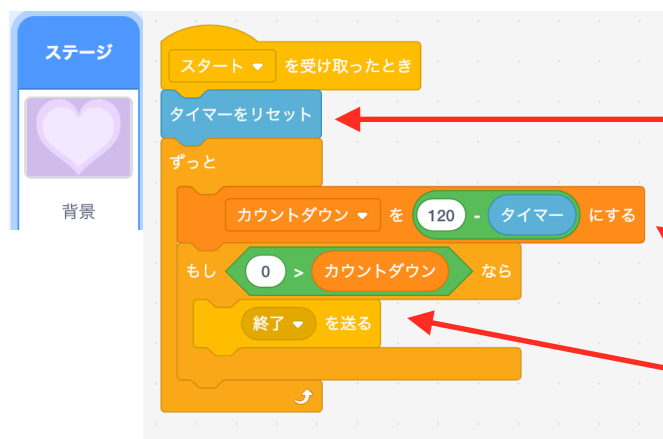
1. 残り時間を表示するために、「カウントダウン」という変数を作成します。



ステージに初期化のブロックを追加します。

変数「カウントダウン」を初期化します。

ステージにカウントダウンのスク립トを追加します。



タイマーはScratchのページを開いた瞬間から作動していますので、ゲーム開始からスタートするためには、スタートの合図を受け取ったら作動していたタイマーを一旦リセットする必要があります。

120-タイマーを変数「カウントダウン」に入れる、という処理を繰り返します。

カウントダウンが0より小さくなったら、メッセージ「終了」を送ります。

ネコに「スタート」のメッセージを追加します。



このメッセージを上記でつくったステージのスク립トで受け取ります。

終了のメッセージを受け取ったときの処理をネコに追加します。



チャレンジ 3.4.c1



リストの中にいくつの要素がはいっているのかは、「○の長さ」のブロックに記録されています。

もしも「順番の長さ」が4なら、リストの中に4つのデータが入っているのでネコに順番を言わせませんが、そうでなければ「全部選んでからスペースキーを押してね!」と言わせます。



赤い枠の中は、元々のプログラムです。