

プログラミング教室のテクノロ



Pythonの道

簡単なプログラム
を作ろう②

もくじ

- ・クイズゲームプログラムを作る



クイズゲームプログラムを作る

変数と条件分岐を使って、クイズゲームを作ろう。

```
print(" サザエさんの旦那さんの名前は？" )
```

．．．．画面に「サザエさんの旦那さんの名前は？」と表示させる

```
答え = input() . . . . 入力を受け付け、入力した文字を「答え」に代入する
```

```
if 答え == " マスオ" : . . . . 「答え」の値が「マスオ」なら
```

```
    print(" 正解です。" )
```

```
else:
```

```
    print(" 不正解です。" )
```

「正解です。」と表示させ、
そうでないなら
「不正解です。」と表示させる。

※変数は通常、アルファベットで定義する。理解が追いつくまで、かな文字で変数を定義してもOK。ただし、他のプログラミング言語ではアルファベットでしか変数を定義できないことは知っておこう。

単語帳 ～英語も一緒に覚えてしまおう！～

■print:印刷する、input:入力する、if:もし~なら、else:その他の

クイズゲームプログラムを作る

「Run Module」またはF5キーを押してプログラムを実行する。

サザエさんの旦那さんの名前は？



マスオ
正解です。

>>>

サザエさんの旦那さんの名前は？



ますお
不正解です。

>>>

if文を理解する

if文を使えば、条件分岐のプログラムを作ることができる。



scratchの
「もし○○ならXX
でなければ△△」
ブロックと同じ

```
if 条件式(○○):      . . . . . もし○○なら    (条件分岐)
print(" XX" )      . . . . . 条件を満たす場合、XXと表示する
else:               . . . . . でなければ    (条件分岐)
print(" △△" )      . . . . . 条件を満たさない場合、△△と表示する
```

※「:」(コロン)の後に改行してインデントを付ける。
インデントの範囲内でしか、条件式は有効とならないことを覚えておこう。

input関数を理解する

input関数はユーザが入力した値を受け取るプログラム



input関数はscratchの「○と聞いて待つ」ブロックと同じ

変数にユーザが入力した値を格納するコード

```
変数 = input(" 値をいれてね" ) . . . . (" ")の中に文字をいれる。
```

※input関数の中で数字を扱う場合、int()をセットで使う。intは数字を扱う場合に使用する。

変数(n)にユーザが入力した数値を代入するコード

```
n = int(input(" 数値を入力してください。" ))
```



数字を入力してください。

クイズゲームプログラムを作る

答えを複数用意してみよう。

「ますお」も正解になるようプログラムを変更しよう。

```
print(" サザエさんの旦那さんの名前は？" )
```

．．．．画面に「サザエさんの旦那さんの名前は？」と表示させる

```
答え = input() . . . . 入力を受け付け、入力した文字を「答え」に代入する
```

```
if 答え == " マスオ" or 答え == " ますお" :
```

```
    print(" 正解です。" )
```

．．．．「答え」の値が「マスオ」または「ますお」なら「正解です。」と表示させ、そうで

```
else:
```

```
    print(" 不正解です。" )
```

ないなら「不正解です。」と表示させる

単語帳 ～英語も一緒に覚えてしまおう！～

■print:印刷する、input:入力する、if:もし~なら、else:その他の

クイズゲームプログラムを作る

「Run Module」またはF5キーを押してプログラムを実行する。



サザエさんの旦那さんの名前は？

マスオ
正解です。

>>>



サザエさんの旦那さんの名前は？

ますお
正解です。

>>>

クイズゲームプログラムを作る

問題数を増やしてみよう。

```
問題 = [  
    ... リスト(問題)に" サザエ..." , " カツオ..." , " サザエ..." を代入する  
    " サザエさんの旦那さんの名前は?" ,  
    " カツオの妹の名前は?" ,  
    " サザエさんが飼っているネコの名前は?" ]  
答え = [" マスオ" , " ワカメ" , " タマ" ]  
    ... リスト(答え)に" マスオ" , " ワカメ" , " タマ" を代入する  
for i in range(3): ... 繰り返し処理 i の値は0⇒1⇒2と変化する  
    print(問題[i]) ... リスト(問題)の0番目、1番目、2番目を順に表示する  
    入力値 = input() ... 変数(入力値)に代入した値を代入する  
    if 入力値 == 答え[i]: ... 入力した値とリスト(答え)の値が等しい  
        print(" 正解です" ) ... 場合、「正解です」と表示する  
    else: ... そうでない場合、「不正解です」と表示する  
        print(" 不正解です" )
```

クイズゲームプログラムを作る



単語帳 ～英語も一緒に覚えてしまおう！～

for:～に向かって、print:印刷する、in:～の中で、range:範囲
input:入力する、if:もし～なら、else:その他の

「Run Module」またはF5キーを押してプログラムを実行する。



サザエさんの旦那さんの名前は？

マスオ

正解です。

カツオの妹の名前は？

ワカメ

正解です。

サザエさんが飼っているネコの名前は？

タマ

正解です。

>>>

リストを理解する（復習）

データをたくさん使うときは「変数」ではなく、「リスト」を使うよ。

リストを作る

新しいリスト

新しいリスト名:

☒ すべてのスプライト
ト用 ☐ このスプライト
のみ

キャンセル OK



scratchの「リスト」ブロックと同じ

リストとは、「番号付きの棚」のようなもので、「何番の中の値」と番号を指定して、値を入れたり、取り出したりする。「名前」ではなく「番号」で指定できるので、データがたくさんあっても扱いやすい。

リストを理解する（復習）

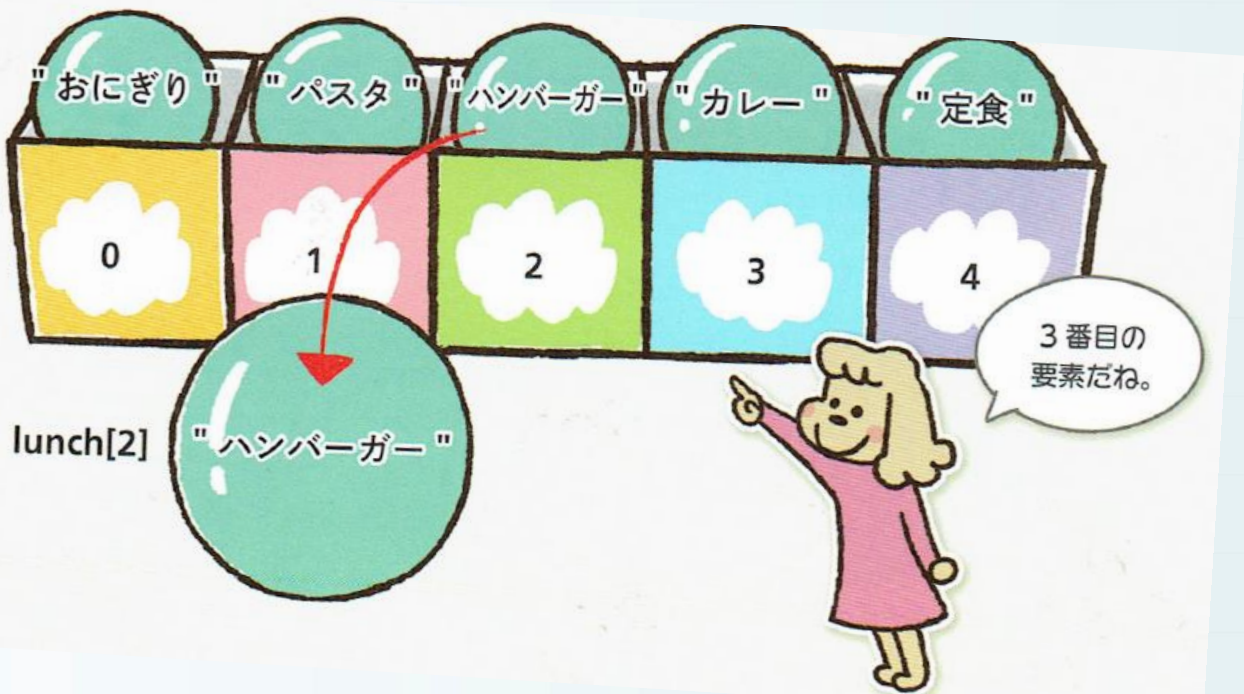
リスト名 = [要素1,要素2,要素3,……] リストを作る
print(リスト名[インデックス]) リストの要素を表示する

```
lunch = [" おにぎり" ," パスタ" ," ハンバーガー" ," カレー" ," 定食" ]  
          0           1           2           3           4  
print(lunch[2])
```

⇒ ハンバーガー
>>>

ポイント

リストはゼロからカウントすることを知っておこう。



for文を理解する（復習）

For文を使えば繰り返し処理を実行することができる。



繰り返し処理とはscratchの「〇回繰り返す」ブロックと同じ

5回繰り返す処理を書くコード（リストを使う場合）

```
lunch = ["おにぎり","パスタ","ハンバーガー","カレー","定食"]  
    . . . . . リスト(lunch)に「おにぎり」、「パスタ」、. . .、「定食」を代入する  
for i in range(5): . . . . . 5回繰り返す処理  
    print(lunch[i]) . . . . . リスト(lunch)から値を順に取り出し、表示  
                        させる
```



おにぎり
パスタ
ハンバーガー
カレー
定食
>>>

クイズゲームプログラムを作る



【問題】

クイズの正解数に応じて得点をカウントできるプログラムを作ってみよう。

サザエさんの旦那さんの名前は？

マスオ

正解です

カツオの妹の名前は？

ワカメ

正解です

サザエさんが飼っているネコの名前は

タマ

正解です

あなたの得点は3点です。

クイズゲームプログラムを作る



【答え】

```
問題 = [  
    " サザエさんの旦那さんの名前は?" ,  
    " カツオの妹の名前は?" ,  
    " サザエさんが飼っているネコの名前は?" ]  
答え = [" マスオ" , " ワカメ," タマ" ]
```

```
スコア = 0
```

```
for i in range(3):  
    print(問題[i])  
    入力値 = input()  
    if 入力値 == 答え[i]:  
        print(" 正解です" )  
        スコア = 1 + スコア  
    else:  
        print(" 不正解です" )  
print("あなたの得点は" , スコア, "点です。")
```

クイズゲームプログラムを作る



【解説】

```
スコア = 0
for i in range(3):
    print(問題[i])
    入力値 = input()
    if 入力値 == 答え[i]:
        print(" 正解です" )
        スコア = 1 + スコア
    else:
        print(" 不正解です" )
```

最初は0

正解すると

正解すると

スコア → 0

→ 1

→ 2

スコア = 1 + スコア

スコア = 1 + スコア

スコア = 1 + スコア

1 = 1 + 0

2 = 1 + 1

3 = 2 + 1

クイズゲームプログラムを作る

問題数を増やして、答えも複数用意してみよう。

```
問題 = [
    ... リスト(問題)に" サザエ..." , " カツオ..." , " サザエ..." を代入する
    " サザエさんの旦那さんの名前は?" ,
    " カツオの妹の名前は?" ,
    " サザエさんが飼っているネコの名前は?" ]
答え = [" マスオ" , " ワカメ" , " タマ" ]
    ... リスト(答え)に" マスオ" , " ワカメ" , " タマ" を代入する
答え2 = [" ますお" , " わかめ" , " たま" ] ... リスト(答え2)に" ますお" ,
スコア = 0 ... スコアに0を代入する          わかめ" , " たま" を代入する
for i in range(3): ... 繰り返し処理 i の値は0⇒1⇒2と変化する
    print(問題[i]) ... リスト(問題)の0番目、1番目、2番目を順に表示する
    入力値 = input() ... 変数(入力値)に入力した値を代入する
    if 入力値 == 答え[i] or 入力値 == 答え2[i]: ... 入力した値とリスト(答え)の値が等しい場合、「正解です」と表示するそうでない場合、「不正解です」と表示する
        print(" 正解です" )
        スコア = 1 + スコア ... スコアに1+スコアを代入する
    else:
        print(" 不正解です" )
```

クイズゲームプログラムを作る

```
print("あなたの得点は" ,スコア,"点です。")
```

単語帳 ～英語も一緒に覚えてしまおう！～

for:～に向かって、print:印刷する、in:～の中で、range:範囲
input:入力する、if:もし～なら、else:その他の

「Run Module」またはF5キーを押してプログラムを実行する。

⇒ サザエさんの旦那さんの名前は？
マスオ
正解です。
カツオの妹の名前は？
ワカメ
正解です。
サザエさんが飼っているネコの名前は？
タマ
正解です。
>>>
あなたの得点は3点です。

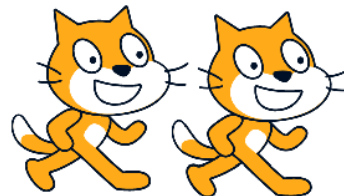
⇒ サザエさんの旦那さんの名前は？
ますお
正解です。
カツオの妹の名前は？
わかめ
正解です。
サザエさんが飼っているネコの名前は？
たま
正解です。
>>>
あなたの得点は3点です。

復習 & チャレンジ

ここまで習ったことをScratchでもできるかチャレンジしてみよう。

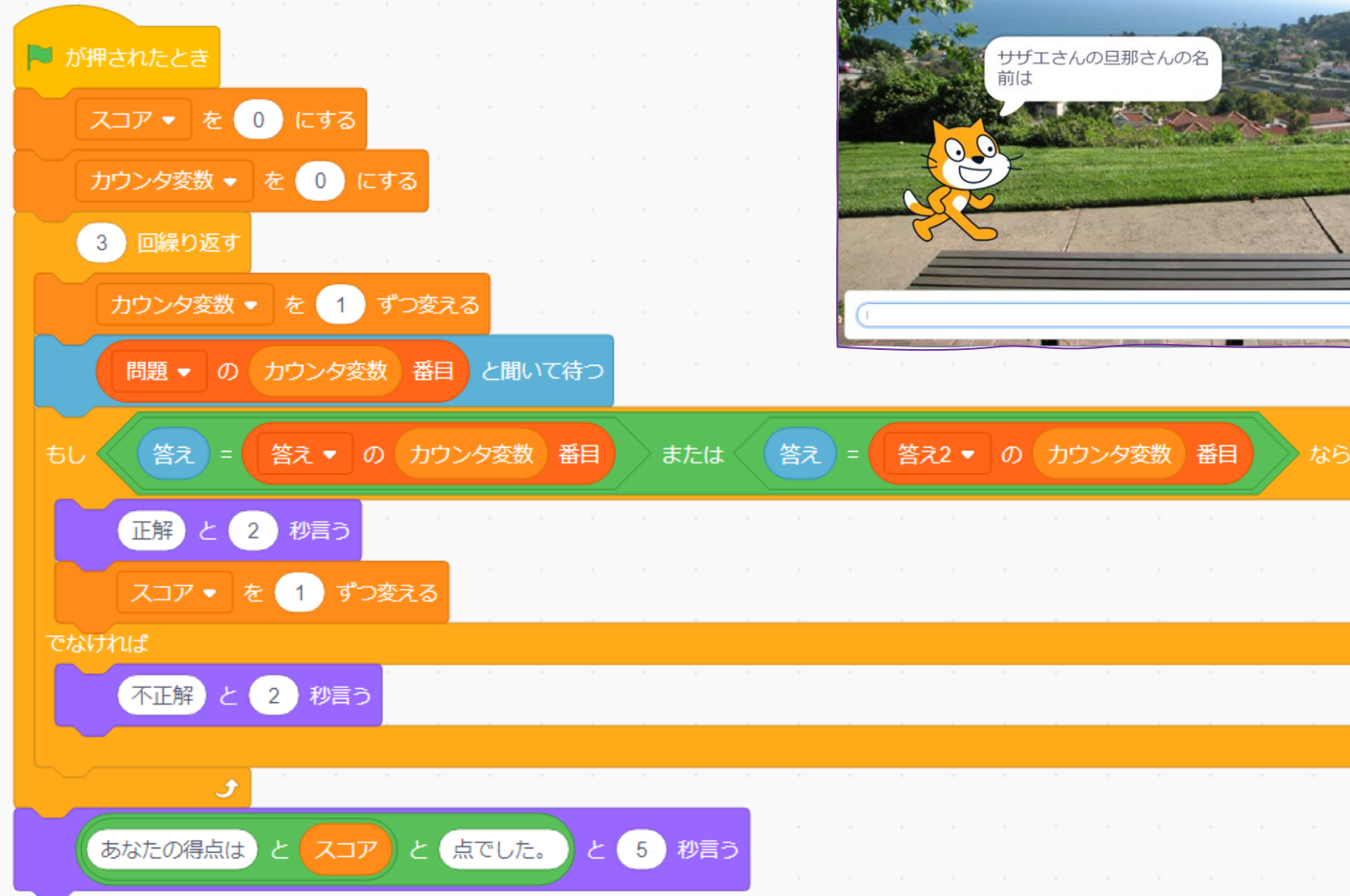
その過程でScratchでできること、Pythonでないとできないことを整理してみよう。

例えば、データの型という概念はscratchにはあっただろうか？



復習 & チャレンジ

scratchでの回答例



メモ



プログラミング教室の テクノロ

なまえ：