Basic stage hint

"・"ごとにヒントをちりばめられると理想です

関数等の説明

Array1obj1

Array1obj2

Array2obj

Basiccalcobj

Ifmazeobj

Outputobj

Tutorial

Array1obj1

・クリア条件

配列の0番目の要素に(配列の長さ-1)の要素を入れる

・配列引数にもつメソッドoperate(array)を作りましょう。

配列名.length()で配列の長さを取り出せます。

配列の0番目にクリア条件の要素を代入しましょう。

Array1obj2

・クリア条件

配列に指定の値を代入する。

・要素番号0には9

・要素番号1には5

・要素番号2には3

・要素番号3には4

・要素番号4には2

・要素番号5には6

Array2obj

・クリア条件

壁の紋章の色を正しい色に切り替える

・idoから配列を手に入れましょう。ido.getArray()で手に入れられるはずです。

・回答用の配列を用意しましょう。

・回答用の配列にidoから手に入れた配列の中身を規則に従い入れていきます

規則はidoの中身が1の時は"on",そうでなければ"off"(文字列)をいれます。

・ido.setResult(回答配列)を行えば井戸からアイテムが手に入るでしょう。

Basiccal

・クリア条件

箱の中身を正しく演算すればアイテムが手に入る

・calc(tubo,oke,hako,tubo\_covered)のメソッドを考えましょう。

・変数を3つ用意してください

[1]壺+桶

[2]壺+箱

[3]壺/箱

を格納します。

・1番初めに[1]に[1]+[2]+[1]\*[2]を代入する

・次に[1]に[1]/[3]\*箱を代入する

・最後に[1]に[1]%桶を代入する

・tubo\_covered.setResult(最後の[1])で覆われた壺にアイテムが出現します。

Ifmazeobj

・クリア条件

正しく馬の自動世話装置を起動する

・care(horse)メソッドを考えましょう。

・馬の状態を知るメソッドはgetHorseState()です。horse.で起動できます。

・馬の世話をするメソッドはcareHorse("care")です。horse.で起動できます。.

・0の時は"feed",1の時は"water",2の時は"wash",3の時は"end"です。

・世話は継続して行うので無限ループの中に置きましょう。

Outputobj

・クリア条件

鍵をすべて見つける

・ヒントを得るためのメソッドを考えます。

output(argument)

・print("Hello,Algeon!!\n")とprintメソッドでは標準出力が可能です

・print("このステージには、", argument , "個の鍵が隠されています\n")と入れてみましょう