

この時間のねらい：① フィールドとメソッドについて学ぼう。

② クラスの考え方と定義の仕方を学ぼう。

<フィールドとメソッド>

前回までに学んだ内容として、変数と関数があります。関数の別名は前回学んだ通り、 といました。変数のほうでは紹介していませんでしたが という名前があります。現実のことをオブジェクトで表現する際、この二つを上手く取捨選択して作成する必要があります。では、どのようなものがどちらなのかを見てみましょう。

(例) (ゲーム自体に関して)

フィールド： 、 、 、
(各自考えられるもの)

メソッド： 、 、 、
(各自考えられるもの)

この他にもたくさんあるはずです。実際、他のゲームで というものがあります。そちらでは、曲自体の編集ができたりもしますが、プロセカではそのようなことはできません。要するに取捨選択により要らない機能とされてしまったわけです。

以上のように自分が作成するものについて、どのような利用者を対象にどのようなものを作成したいかを考えることでフィールド、メソッドを限定していくことができます。しっかりとそれらを決めることで、 への対処や、プログラムの難易度を軽くしたり、逆に重くしたりもできます。

また、最初に出た という言葉ですが、これはフィールド、メソッドの集合体だと思ってください。フィールドとメソッドをひとまとめにして、それを呼び出すことにより、同じ処理をくり返すというものを、今回は学んでいこうと思います。

このようにオブジェクトを作成して、一度作成したものを再利用したり、作成については情報を共有しておけば別の人と協力して作成などが実現できます。

この考え方を と言います。

< _____ >

オブジェクトを使用するためには、その前に _____ を定義する必要があります。これはオブジェクトの _____ だと思ってください。例えば、ある図形を描くオブジェクトについて、色の変更を前回の関数のように、引数を使用して設定できれば、とても効率的になります。

では、この例を見てみましょう。前回と同じく「サンプル」の「Ex_10_01」を開いてください。この中の 14 行目以降がクラスの定義です。

その中でも、16 行目から 19 行目は _____、27 行目から 34 行目は _____ があるので、フィールドとメソッドを使用した設計図になります。

クラス(設計図)の作成が終了したら、実行するための準備をします。例えば、1 行目では _____ の宣言を行っています。また、new を使用することでオブジェクトを生成(_____)します。ここまでの、見た目は何とかになります。

ただ、作成したメソッドを使用する場合は、
“ _____ ”のように draw()メソッド内に記述すると、
メソッドを適用させることができます。

また、Processing にはタブ機能(分割機能)があります。プログラム記述場所の上のところに逆三角形マークの付いた部分があります。ここをクリックすると、「新規タブ」と出てきます。クリックすると名前を付けるウィンドウが開き、そこに作成したいものの特徴などを書いておきます。(頭文字は _____ で必ず _____ での記入をします。)

そこで分けて書いたものが「Ex_10_02」です。

このときのポイントはタグ名とクラス名を一致させることです。これをしないと本当にプログラミングを学んだかどうか疑われてしまいます。

とりあえず、サンプルの内容を書き換えることで、クラスの作成を試みましょう。