

この時間のねらい：① 条件分岐について学ぼう。

今回は前回に引き続き 3 大処理について、前回以外の 2 つについて学びます。  
 まずは変数によって処理を変更する \_\_\_\_\_ について学びましょう。基本的に構文は  
 ほかのプログラミング言語でも同じもの（だと思う）なので、この構文をしっかりと使える  
 ようにしましょう。

#### ◇条件分岐の使われている例

そもそも、条件分岐の使われている物にはどんなものがあるでしょうか。それは以下  
 のものです。

- ・ \_\_\_\_\_ (前回の授業内容より)
- ・ \_\_\_\_\_ (ポケモンより)
- ・ \_\_\_\_\_ (アプリ版カードゲームより)
- ・ \_\_\_\_\_ (3D、ブロック崩しより)

今挙げたものはほんの一部の例です。これらを実装する具体的なやり方を学びます。

今回の内容では、以前紹介した“ \_\_\_\_\_ ”が理解できていないと難しいです。復習しながら  
 学びましょう。

#### ◇復習 (変数)

変数とは、前の例では箱を用意すると言いました。基本的には変数に値を代入して使いま  
 す。このことを別の言い方として \_\_\_\_\_ という言い方もします。

この \_\_\_\_\_ が非常に重要です。(主にバグ修正のときに手がかりとなります。)

また、変数を \_\_\_\_\_ しておかないとエラーが起きるため注意です。

それでは実際にどのようにやるのかを見ていきましょう。(裏面へ続く)

### 分岐処理の実装方法

分岐のための関数は右のものです。この単語、英語の時間に習いませんでしたか？

if は \_\_\_\_\_ という仮定法のところで出てきます。

else は \_\_\_\_\_ という意味です。基本的に引数のところに条件を書き、それに該当する場合はその後ろの \_\_\_\_\_ 内の処理を行うという流れです。

```
int x = 1;
if(x == 1){
    println(1);
}else if(x == 2){
    println(2);
}else{
    println("該当なし");
}
```

また、if 文以外も方法があります。右の画像を見てください。switch 文です。この関数でやっていることは引数内の変数がどのケースかによって、処理方法を変更というやり方になります。

```
int x = 1;
switch(x){
    case 0: println(0); break;
    case 1: println(1); break;
    case 2: println(2); break;
}
```

ただし、それぞれの case 内の最後には break; を入れなければしっかりと処理がされなくなるため、注意してください。

以上のどちらか一方で条件分岐を行うようにしてください。