

Wordpress + PHP

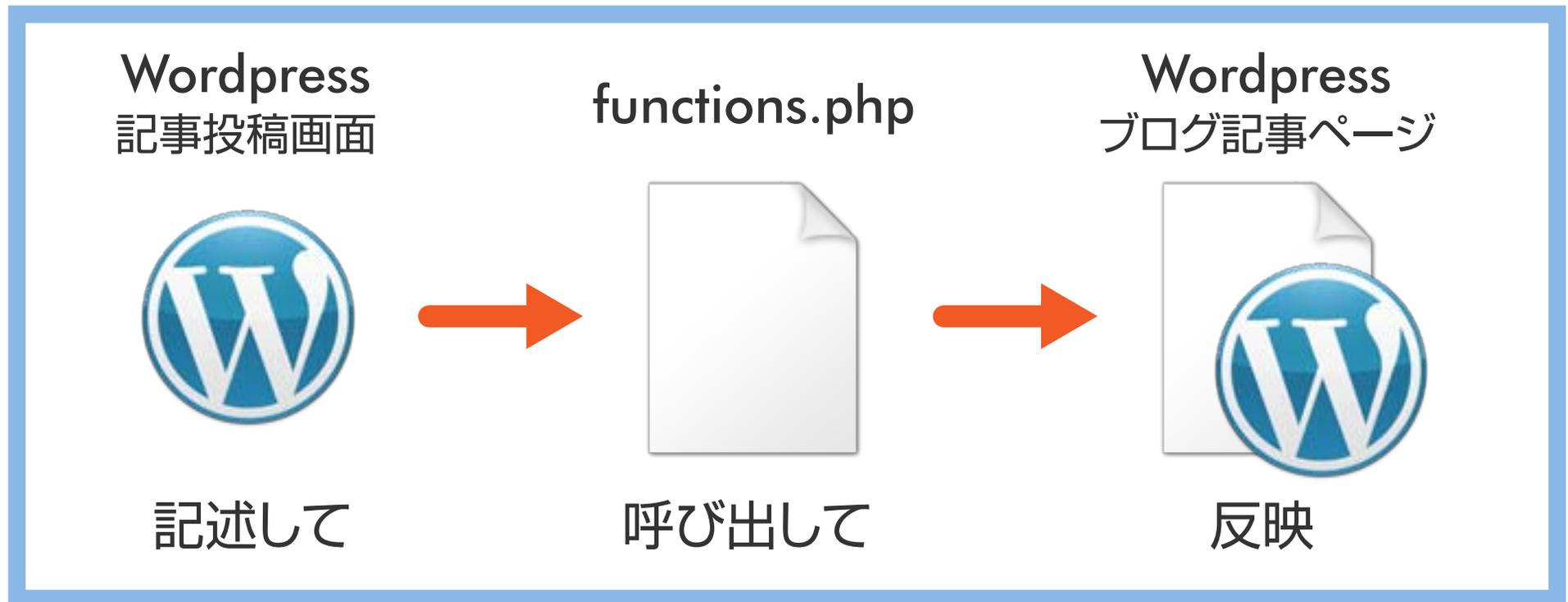
～WPショートコードについて～



SaCSS Vol.24
2011/05/14(土)

Wordpress ショートコードとは

記事投稿に記述して使用するWordpress独自関数。
ある処理をfunctions.phpに記述しておき、ショートコード(独自タグ)で呼び出して起動させます。



ショートコードの種類

ショートコードは[](ブラケット)で関数名(自由に名称設定できる※)を囲います。単独で使用するものと、前後を囲う2種類があります。

- 自己完結型ショートコード(単独で使用)

[hogehoge]

- 囲み型ショートコード(HTMLタグの様に前後を囲む)

[hoge]テキストなど[/hoge]

※Wordpressで使用している名称を付けるとエラーが出てしまうので注意

基本は
HTMLと
同じ

ショートコードの記述方法

ショートコードは下記のような感じでfunctions.phpに記述します。PHPなどの関数と似ています。

functions.php



■ショートコード記述例

```
function shortTest() {  
    return "ショートコードのテスト";  
}  
add_shortcode('stest', 'shortTest');
```

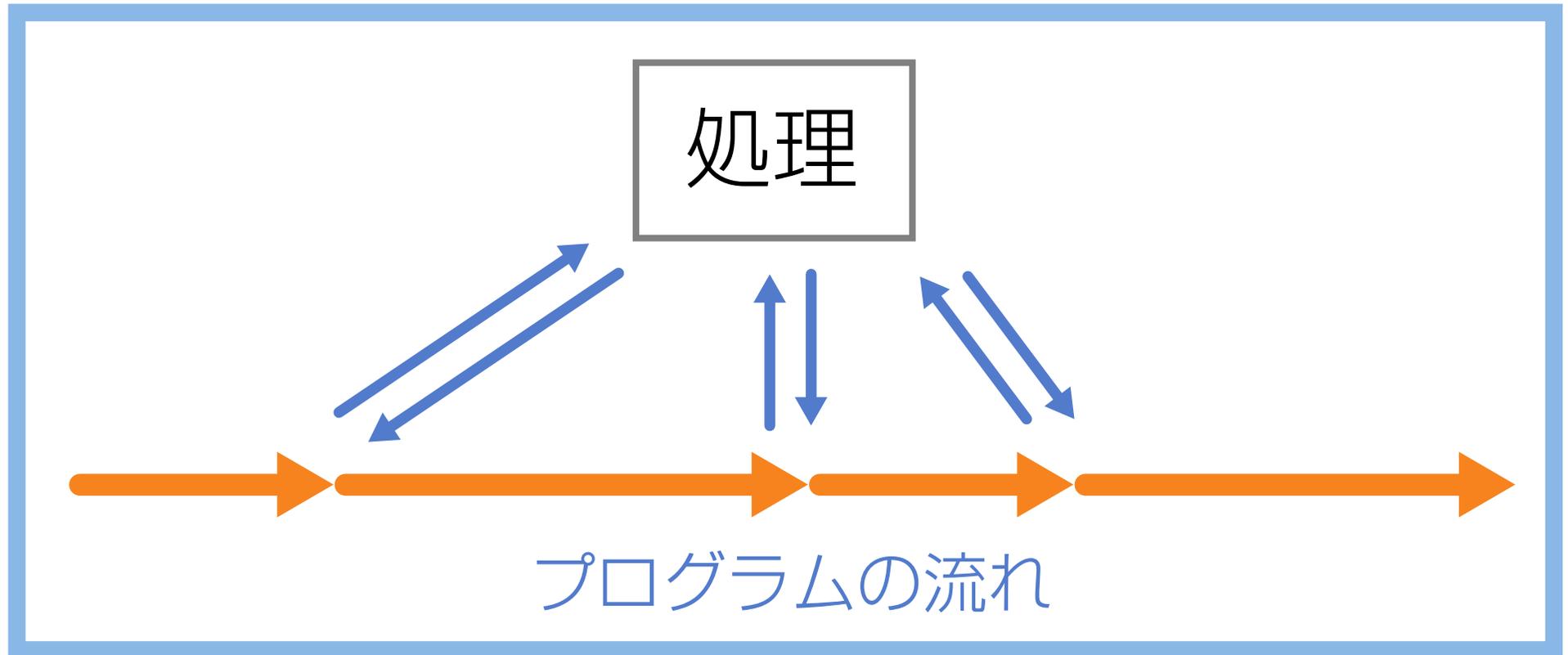
関数とは？

関数とは、プログラム内にある同じ処理をまとめ、外部ファイルから呼び出すことでソースの整理、記述ミスなど効率化を図ることが出来ます。



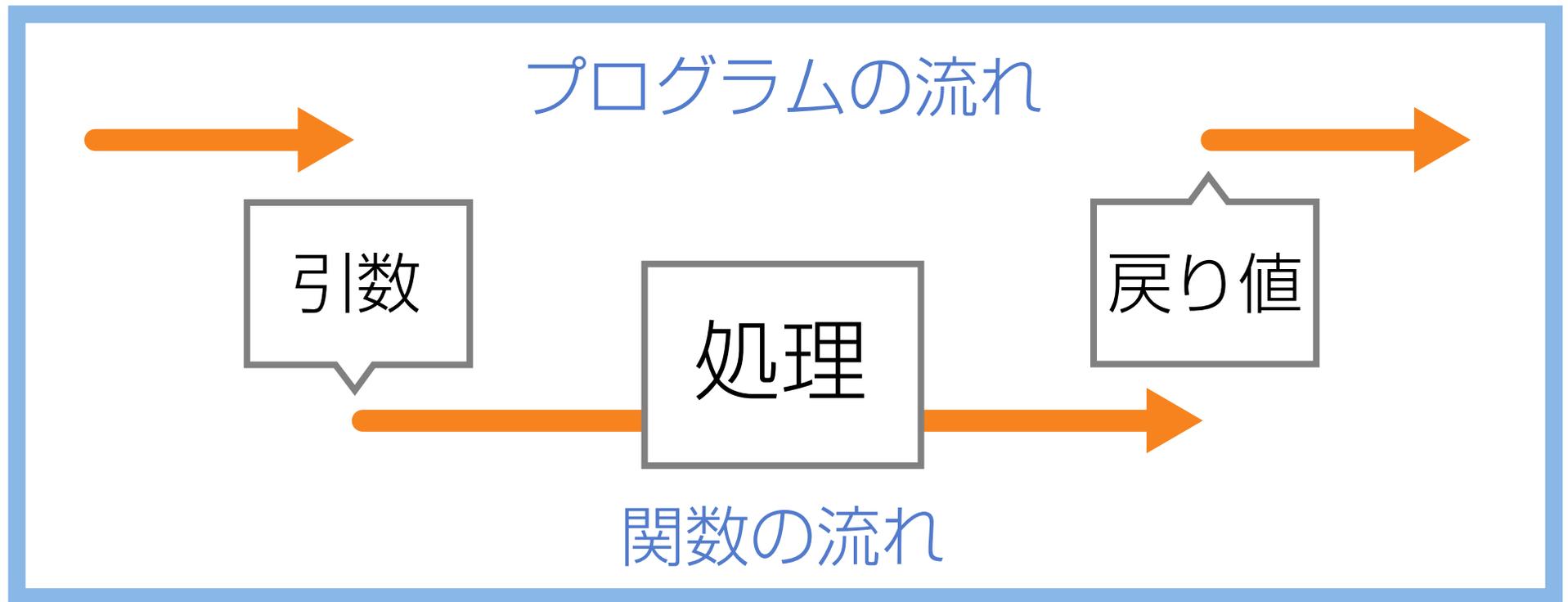
関数とは？

関数とは、プログラム内にある同じ処理をまとめ、外部ファイルから呼び出すことでソースの整理、記述ミス回避など効率化を図ることが出来ます。



関数とは？

また、関数は値(引数)を受け、処理を行い、それに基づいた結果の値(戻り値)を返すといった、特定の計算処理に適しています。



関数とは？

関数は、functionで定義の宣言を行い、行う一連の処理に対して名称を付けます。基本的にreturnで関数処理を終える形となります。

■関数の構成

```
function 関数名(引数) {  
    行う処理  
    return 戻り値;  
}
```

記述は
基本
ほぼ同じ



ショートコードの記述方法

関数の解説を踏まえ、ショートコード記述例をもう一度見てみましょう。

functions.php

■ショートコード記述例

```
function shortTest() {  
    return "ショートコードのテスト";  
}  
add_shortcode('stest', 'shortTest');
```

ショートコードの記述方法

前ページの記述例を解説するとこんな感じです。

関数記述の宣言

```
function shortTest() {
```

{ }内に処理情報を記述する

```
    return "ショートコードのテスト";
```

returnは、戻り値

```
}
```

関数名を同期させておく

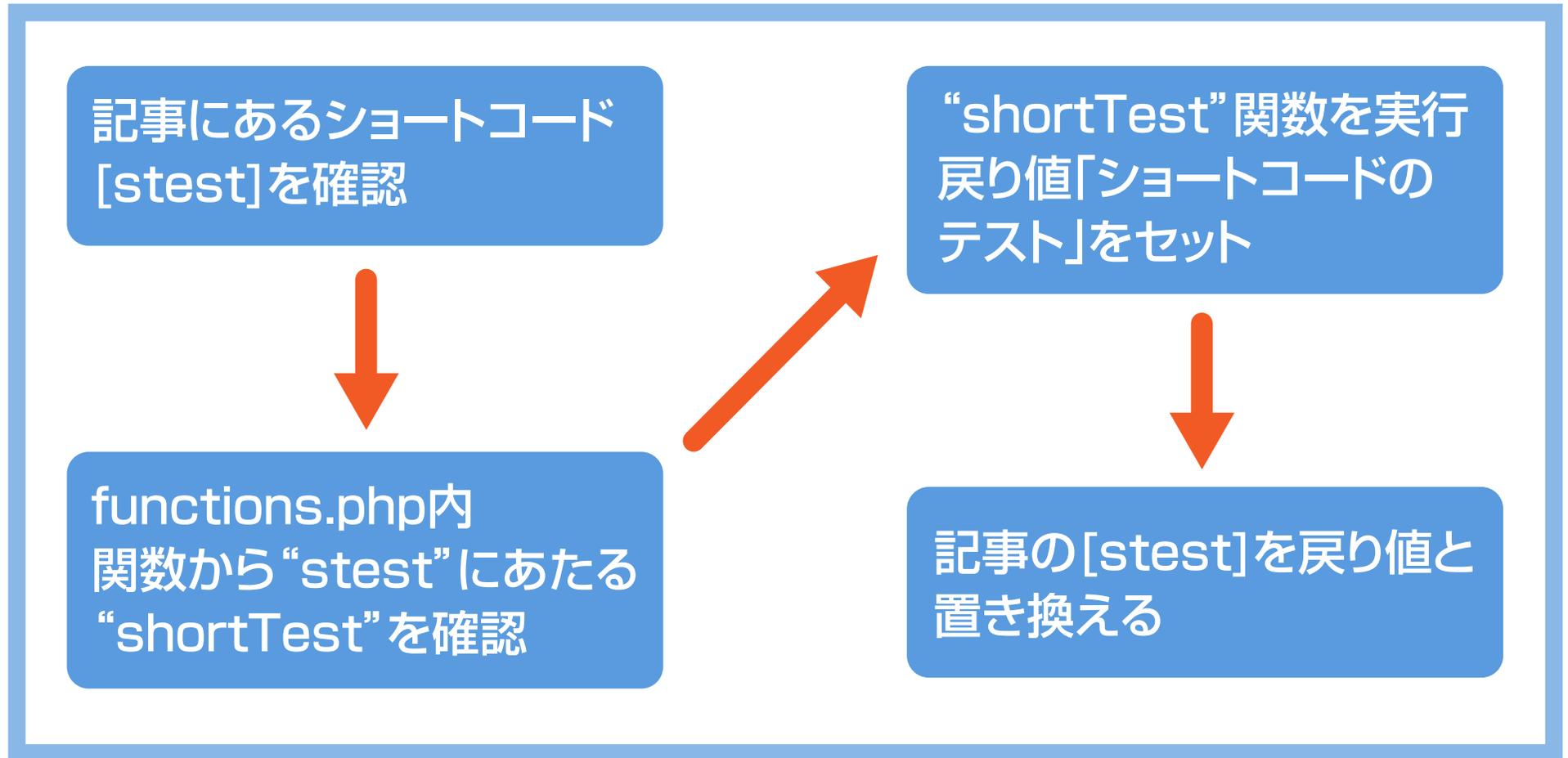
```
add_shortcode('stest', shortTest);
```

ショートコードを追加する関数

ショートコード用の名称

ショートコードのフロー

状況をまとめてみるとこのような感じになります。



ショートコードの引数

ショートコードは引数も受け渡すことが可能です。

■引数を使ったショートコード記述例

```
function hikisu($atts) {  
    extract(shortcode_atts(array(  
        'num' => 0,  
    ), $atts));  
    return $num * 10;  
}  
add_shortcode('hiki', 'hikisu');
```

ショートコードの引数

これも説明すると…。

\$attsは属性の配列
attributes

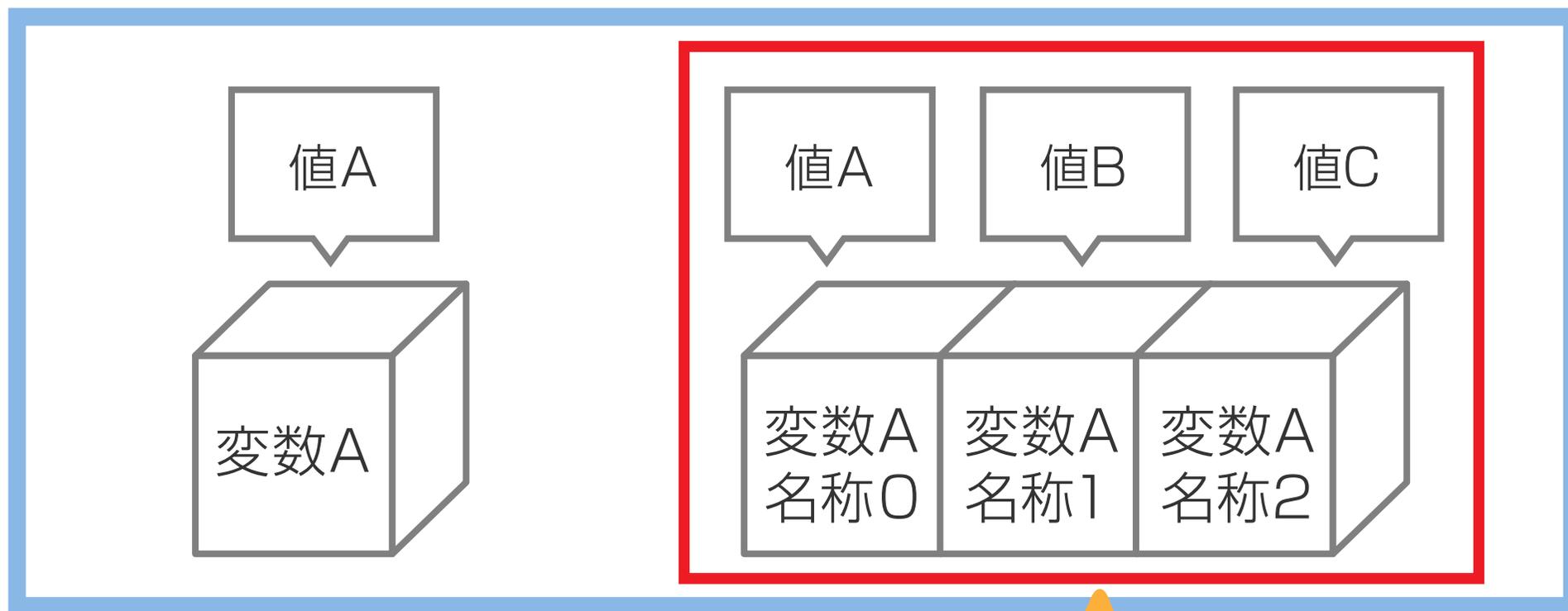
ここら辺
次ページ
で解説

```
function hikisu($atts) {  
    extract(shortcode_atts(array(  
        'num' => 0,  
    ), $atts));  
    return $num * 10;  
}  
add_shortcode('hiki', 'hikisu');
```

変数\$numに10倍した
ものを戻り値として利用

array関数の解説

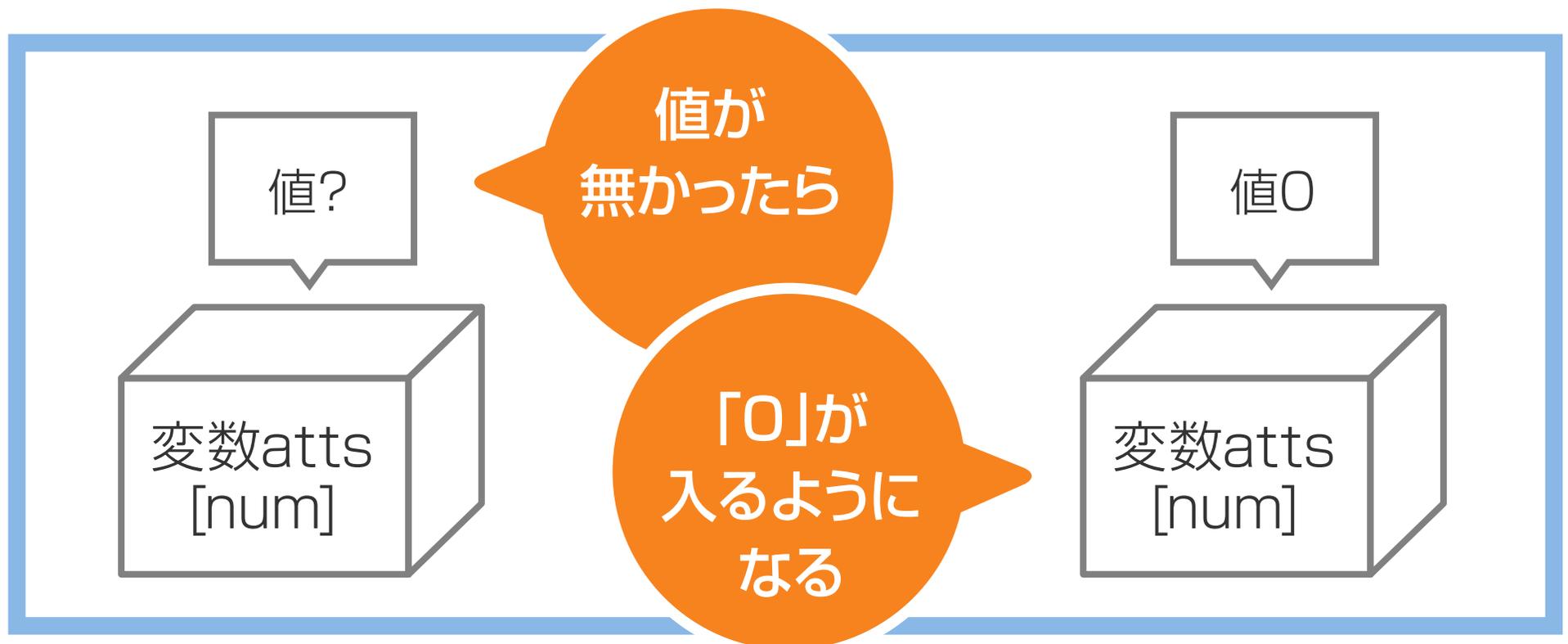
変数には値が一つ入るのが基本です。変数に名称を付け、連想配列化した場合にはその名称ごとに値を入れることができます。array関数は一括で値を入れることができる関数です。



値をまとめて設定できる

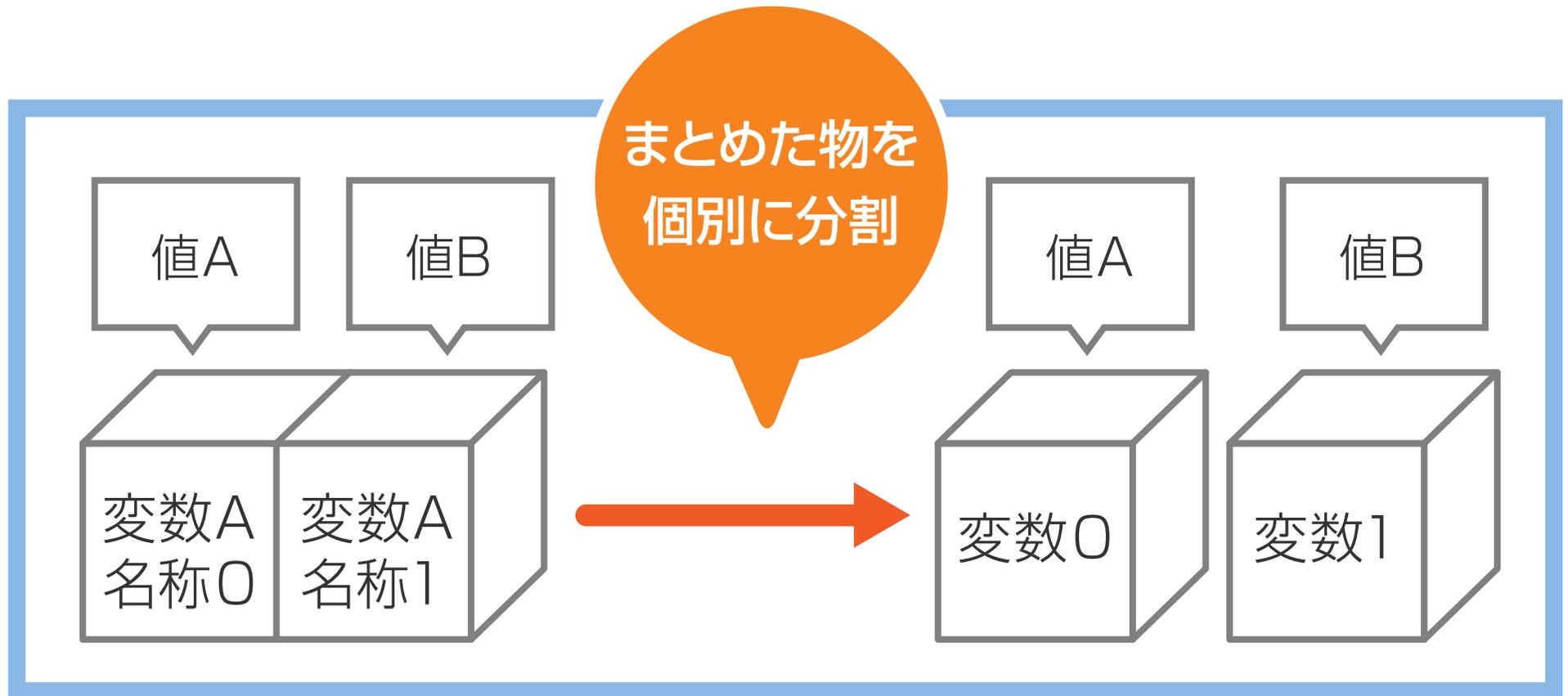
shortcode_atts関数の解説

shortcode_attsは、属性値の値が無かった場合、基本値を決め、代入するWordpressの関数です。今回のケースでは、numの値が無い場合numの値が0になります。



extract関数の解説

extract関数は、連想配列に付けた名称を個別の変数とし、対応した値を代入します。



引数を使った結果

最終的に、記事投稿にショートコードを記述する際、属性値を追加記入できるようになり、その属性値を元に処理と結果を表示させます。

■hiki単体の場合

numに値「0」が入るので、
num×0=0となり、[hiki]には0が表示される

■hikiの属性numに10を入れた場合

numに値「10」が入るので、num×10=1000となり、
[hiki]には1000が表示される

属性の
値により
結果が変化

囲み型ショートコード

囲み型ショートコードの場合では、下記のような感じで functions.php に記述します。

■ 囲み型ショートコード記述例

```
function kakomi( $atts, $content = null ) {  
    extract( shortcode_atts( array(  
        'class' => 'kihon',  
    ), $atts ) );  
    return '<p class="' . $class. '">' . $content . '</p>';  
}  
add_shortcode('kako', 'kakomi');
```

囲み型ショートコード

囲み型ショートコードの場合、ショートコードに第二引数を設定することで囲み型となります。

nullを代入して
中身をなしにしている

■ 囲み型ショートコード記述例

```
function kakomi( $atts, $content = null ) {  
    extract( shortcode_atts( array(  
        'class' => 'kihon',  
    ), $atts ) );  
    return '<p class="' . $class. '">' . $content . '</p>';  
}  
add_shortcode('kako', 'kakomi');
```

本文と置き換える

テンプレート内でショートコード

ショートコードは投稿記事内で使用するものですが、テンプレートなどでも下記のように記述することで起動させることができます。

■テンプレート内でショートコードを使う場合の記述例

```
<?php echo do_shortcode(' [tag] '); ?>
```

文字列に
含まれる
ショートコード
を変換する

ご清聴
ありがとう
ございました。

