第２回　配列・連想配列・HTMLでの画面出力・foreach

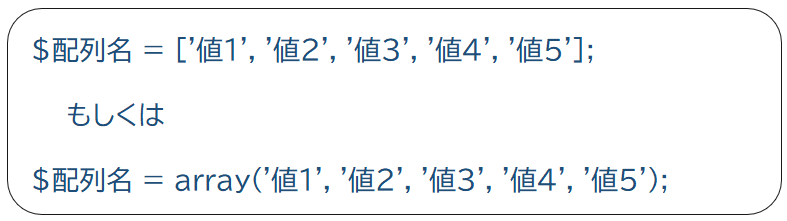
２－１．一次元配列

配列は、関連性のあるデータをまとめておきたいときに使用します。

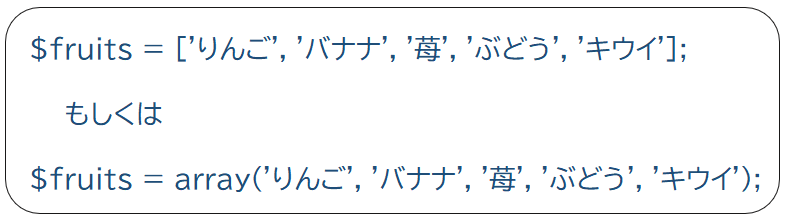
２－１－１．一次元配列の宣言と初期化

一次元配列の宣言と初期化は、下記のように記述します。

■構文

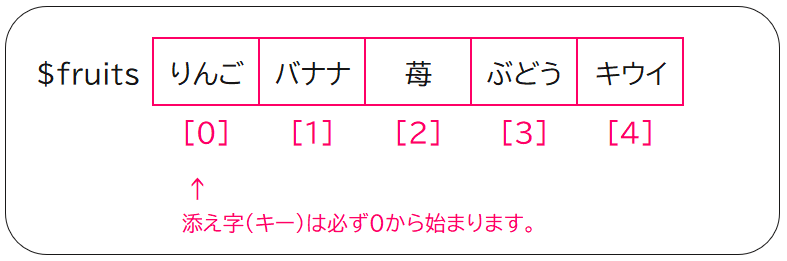


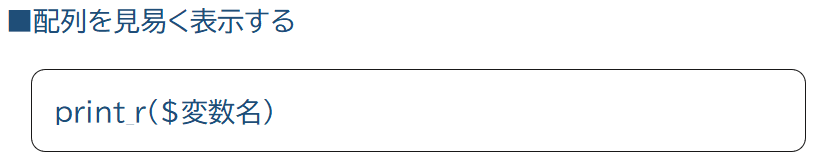
■宣言例



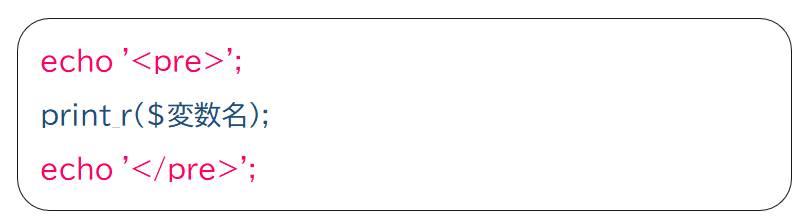
作成された配列 $fruitsは次のようなイメージです。

■配列 $fruits イメージ



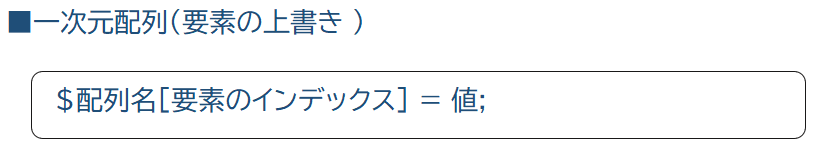


ただし、print\_rだけでは、配列の内容が1行で出力されますので、前後に<pre>タグを出力して、改行による整形を追加しましょう。



グラフィカル ユーザー インターフェイス, テキスト, アプリケーション

自動的に生成された説明



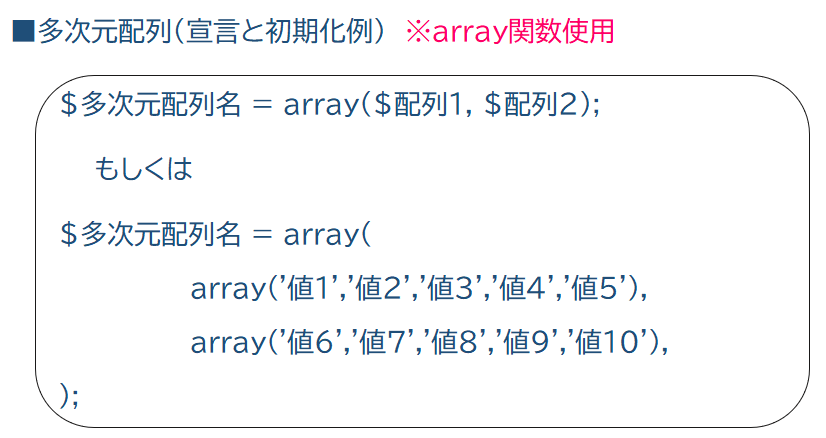
[ ]の中にインデックス番号を指定すると、既にある値が上書きされます。

|  |  |
| --- | --- |
| 警告 単色塗りつぶし | もし、その時点で存在しない要素数（インデックス番号）を指定した場合、  JavaではIndexOutOfBoundsExceptionが発生しますが、  PHPでは、エラーは発生しません。指定したインデックス番号の値として、新たに追加されます。  【例】  $fruits = ['りんご', 'バナナ', '苺', 'ぶどう', 'キウイ',’パイナップル’];  $fruits[8]=’スイカ’  【実行結果】print\_rで表示したとき  Array  (  [0] => りんご  [1] => バナナ  [2] => 苺  [3] => ぶどう  [4] => キウイ  [5] => パイナップル  [8] => スイカ  )  【注意】  インデックス番号[6][7]を飛ばして、[8]に追加となりますので、下記のように[7]を表示しようとした場合、  echo $fruits[7],'<br>';  [7]は存在しないので、実行時エラーとなります。 |

２－２．多次元配列

２－２－１．多次元配列の宣言と初期化

多次元配列に値を代入するには、array関数の使用、もしくは短縮構文での記載が可能です。



■SampleCode１

$name = array(‘ECC太郎’,‘ECC次郎’,’ECC三郎’); //配列１

$class = array(‘1A’,’2B’,’3A’); //配列２

$list = array($name, $class); //多次元配列(配列1と2を使用)

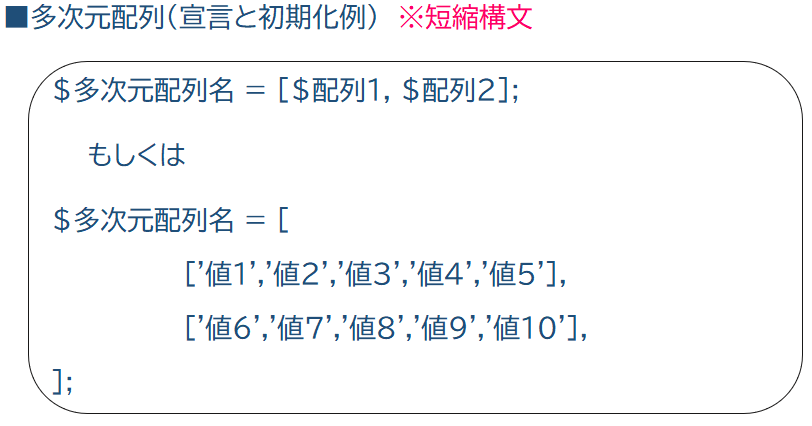
■SampleCode２

$subject = array(

array(‘英語’,’数学’,’理科’),

array(‘English’,’Math’,’Science’)

);



■SampleCode１

$name = [‘ECC太郎’,‘ECC次郎’,’ECC三郎’]; //配列１

$class = [‘1A’,’2B’,’3A’]; //配列２

$list = [$name, $class]; //多次元配列(配列1と2を使用)

■SampleCode２

$subject = [

[‘英語’,’数学’,’理科’],

[‘English’,’Math’,’Science’]

];

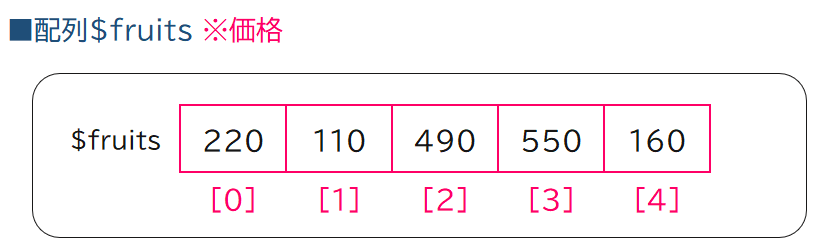
２－３．連想配列・foreach

２－３－１．連想配列とは

多次元配列では、要素のインデックス番号は数字の0から順に自動的に割り当てられていました。

しかし、数字ではなく、文字列の方が直感的に分かり易い場合もあります。

例えば、課題２－２までで使用した$fruitsですが、今までは名前を保存していましたが、それぞれの価格を保存したい場合、下図のような格納方法では、どれが何の値段かが分かりません。



こういった場合に、要素名、すなわちキーを設定できる連想配列が便利です。



■SampleCode１

$list = [

‘name’ => ‘ECC太郎’,

‘class’ => ‘1A’

];

■SampleCode2

$list = array(

‘name’ => ‘ECC太郎’,

‘class’ => ‘1A’

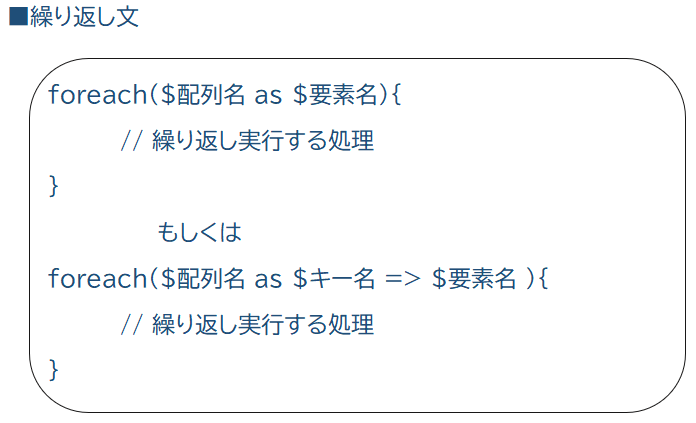
};

通常の配列と同じように、array関数や短縮構文で宣言しますが、特徴として、キーは文字列なので、「”」（ダブルコーテーション）で囲み、「=>」（ダブルアロー演算子）の後に、値を記述します。値は、文字列の場合は囲む必要がありますが、数値の場合はそのまま記述します。

２－３－３．繰り返し文との併用（foreach）

今まで変数や配列の内容はprint\_rやvar\_dumpで表示してきましたが、このような表示では、見た目を整えることができません。Webページでは、配列の一つ一つをHTMLタグに囲んで見た目を整えるといった使い方をします。そのような時に繰り返し文を使うと便利です。

今回はまだ、HTML部分は記載しませんが、慣れるために、まずは配列の要素を繰り返し文で表示してみましょう。



■SampleCode

＄hoge = [‘あいうえお’,’かきくけこ’,’さしすせそ’,’たちつてと’];

foreach($hoge as $h) {

echo $h, ‘<br>’;

}

■SampleCode実行結果

あいうえお

かきくけこ

さしすせそ

たちつてと

２－４．HTMLでの画面出力

２－４－１．HTMLの使用

前項までは、PHPのecho文にて画面出力をしていました。

本項からは画面出力にはHTMLを使用しましょう。

HTML部分は、下図のように<?php ?> 部分の後に記載します。

■全体構造

**<?php**

**// ここにコードを記述**

**?>**

**<!DOCTYPE html>**

**<html lang="ja">**

**<head>**

**<meta charset="UTF-8">**

**<meta http-equiv="X-UA-Compatible"**

**content="IE=edge">**

**<meta name="viewport"**

**content="width=device-width, initial-scale=1.0">**

**<title>Document</title>**

**</head>**

**<body>**

**<!—ここにデザイン部を表記 -->**

**</body>**

**</html>**

サンプルコードで変数の内容をHTMLで表示します。

■Sample（変数の宣言）　※<?php ?>　内での宣言

テキスト

中程度の精度で自動的に生成された説明

タグの中に、変数の値のみを表示する場合、**<?= $変数名 ?>**で表示することができます。

■Sample（HTMLデザインに変数を埋め込む）

<!DOCTYPE html>

<html lang="ja">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>HTMLによる画面表示</title>

</head>

<body>

    <h1>HTMLでの表示</h1>

    <!-- 第1段階 -->

    <h2>php echoでの表示</h2>

    <?php echo "<p>ID:{$id}</p><p>名前：{$name}</p><p>クラス：{$homeClass}</p>"; ?>

    <!-- 第2段階 -->

    <h2>タグにPHP変数埋め込み</h2>

    <p>ID：<?= $id ?></p>

    <p>名前：<?= $name ?></p>

    <p>クラス：<?= $homeClass ?></p>

</body>

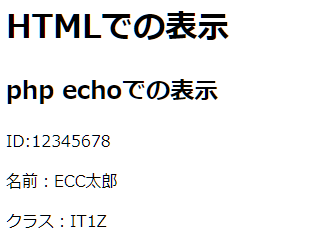
</html>

**■Sample実行結果**

それぞれの項目をpタグ（：以降は変数の値を出力）

h2タグ

h1タグ



２－４－２．デザインの追加

前期ではCSSファイルを自作していましたが、今回はBootstrap5というフレームワークを用いて、デザインを整えてみましょう。

1. 下記サイトにアクセスします。

<https://getbootstrap.jp/docs/5.0/getting-started/download/>

1. コンパイルされたCSSとJSをダウンロードします。下図の赤枠押下

グラフィカル ユーザー インターフェイス, アプリケーション

自動的に生成された説明

1. ダウンロードしたzipファイルを展開し、中にある「css」「js」フォルダを課題ファイルと同じ階層に配置してください。（今回課題で使用するのはCSSのみです。JSは自由に使用してみてください）

グラフィカル ユーザー インターフェイス, テキスト

自動的に生成された説明

1. CSSフォルダ内の「bootstrap.min.css」というスタイルシートを使用します。

次回の課題からは「bootstrap.min.css」を読み込んで作成してください。

■Bootstrap定義済みクラス一例

|  |  |
| --- | --- |
| class | 用途 |
| container-fluid | 可変幅のコンテナを設定する |
| lead | 見出し行ではなく、通常の段落内のテキストを目立たせる |
| small | small要素と同じスタイルを適用できる |
| table | 各テーブルクラスを使用するためのクラス |
| table-striped | 奇数行の背景色と文字色を設定する |
| table-hover | テーブル行にマウスオーバー表示を設定する |
| table-bordered | テーブルの罫線をつける |
| table-dark | テーブルを下記の設定にする  ・背景色#373b3e  ・文字色#fff |