

PHP : Funcions i estructures de dades

Funcions

Una funció és un bloc de codi que es pot reutilitzar.

Les variables que es declaren dins una funció són locals i són inaccessibles des de fora d'aquesta funció.

PHP té moltes funcions **predefinides**, per exemple la funció «date» retorna la data i hora actuals i es genera segona el format especificat.

```
<?php
    echo date("d/m/Y H:i:s");
?>
```

També es poden crear funcions **personalitzades**.

```
<?php
function saludar($nom) {
    return "Hola, " . $nom . "!";
}

echo saludar("Joan"); // Crida la funció amb el nom "Joan"
?>
```

En aquest cas, la funció té un paràmetre d'entrada i retorna un string.

Funció de **callback**: Funció que es passa com a argument a una altra funció. La funció que rep el callback invoca aquesta funció en algun moment durant la seva execució. Aquesta tècnica et permet personalitzar el comportament d'una funció o reaccionar a esdeveniments específics.

```
<?php
function processarNom($callback, $nom) {
    // Crida la funció de callback amb la variable passada
    echo $callback($nom);
}

function saludar($nom) {
    return "Hola, " . $nom . "!";
}

// Crida a processarNom passant saludar com a callback i el nom "Joan"
processarNom('saludar', 'Joan');
?>
```

S'usen habitualment amb funcions de manipulació d'arrays com **array_filter()**, per filtrar els elements de l'array, o **array_map()**, per aplicar la funció de callback a cada element de l'array.

Arrays

Els arrays són de tipus compost, permeten emmagatzemar múltiples valors en una sola variable.

Arrays **indexats**: utilitzen números enters com a claus, per defecte, les claus comencen a comptar des de 0.

```
<?php
$fruites = ["Poma", "Plàtan", "Mango"];
echo "M'agrada menjar " . $fruites[0] . " i " . $fruites[2] . ".";
// Afegir un element (al final)
$fruites[] = "Raïm";
// Eliminar un element (índex 1)
unset($fruites[1]);
?>
```

Arrays **associatius**: Un array amb claus personalitzades (strings).

```
<?php
$edats = ["Joan" => 25, "Maria" => 22, "Pere" => 30];
echo "Maria té " . $edats['Maria'] . " anys.";
// Afegir un parell clau-valor
$edats["Ramon"] = 55;
// Eliminar un element (el parell amb clau Maria)
unset($edats["Maria"]);
?>
```

Arrays **multidimensionals**: Un array que conté altres arrays dins seu. Això permet estructurar les dades en diverses dimensions.

```
<?php
$notesEstudiants = [
    "Joan" => ["Matemàtiques" => 8, "Ciències" => 7, "Història" => 9],
    "Maria" => ["Matemàtiques" => 6, "Ciències" => 8, "Història" => 7],
    "Pere" => ["Matemàtiques" => 9, "Ciències" => 6, "Història" => 8]
];
echo "Nota Joan en matemàtiques: " . $notesEstudiants["Joan"]["Matemàtiques"];
?>
```

Bucles

Els bucles són útils per repetir un bloc de codi diverses vegades.

```
<?php
    $i = 0;
    while ($i <= 5) {
        echo "Número: $i <br>";
        $i++;
    }
?>

<?php
    for ($i = 0; $i <= 5; $i++) {
        echo "Número: $i <br>";
    }
?>

<?php
    $i = 0;
    do {
        echo "Número: $i<br>";
        $i++;
    } while ($i <= 5);
?>
```

Bucles per iterar arrays indexats.

Sintaxi → `foreach (array as elementIterat):`

```
<?php
    for ($i = 0; $i < count($fruites); $i++) {
        // Amb isset ens assegurem que la clau existeix abans d'accedir-hi
        if(isset($fruites[$i])) echo $fruites[$i] . "<br>";
    }
    foreach ($fruites as $fruita) {
        echo $fruita . "<br>";
    }
?>
```

Bucles per iterar arrays associatius.

Sintaxi → `foreach (array as key => value):`

```
<?php
    foreach ($edats as $nom => $edat) {
        echo "$nom té $edat anys.<br>";
    }
?>
```

Bucles per iterar arrays multidimensionals.

Sintaxi → `foreach (array as key => subArray)`

`foreach (subArray as subKey => value)`

`// Codi a executar amb subKey i value`

```
<?php
// Mostrar les notes de Joan
echo "Notes d'en Joan:<br>";
foreach ($notesEstudiants["Joan"] as $materia => $nota) {
    echo "$materia: $nota<br>";
}

// Iterar sobre l'array multidimensional
foreach ($notesEstudiants as $nom => $notes) {
    echo "Notes de " . $nom . "<br>";
    foreach ($notes as $assignatura => $nota) {
        echo $assignatura . ": " . $nota . "<br>";
    }
    echo "<br>";
}
?>
```

Resum pràctic

Pots consultar aquest fitxer HTML amb codi PHP integrat corresponent als conceptes que has vist en aquest document:

<https://github.com/xbaubes/DesenvolupamentWeb/blob/main/Backend/PHP/PHP-algorismica/algorismica.php>

ACTIVITATS

Les dades en forma d'arrays es troben en el següent fitxer:

<https://github.com/xbaubes/DesenvolupamentWeb/blob/main/Backend/PHP/PHP-algorismica/algorismicaActivitats.php>.

Crea un fitxer de nom «funcions.php» on crearàs totes les funcions personalitzades per tractar les dades d'aquestes activitats. Les teves funcions s'han de cridar des del fitxer adjuntat.

Aquí pots consultar una llista amb totes les funcions pròpies dels arrays:

<https://www.php.net/manual/es/ref.array.php>

1. Treballarem amb el següent array de ciutats catalanes:

```
<?php
$ciutatsCatalunya = [
    "Barcelona",
    "Girona",
    "Lleida",
    "Tarragona",
    "Badalona",
    "Terrassa",
    "Sabadell",
    "Reus",
    "Santa Coloma de Farners",
    "Mataró"
];
?>
```

Crea una funció que li passis un array i retorni un nou array que contingui només els elements amb índex parell, concatena el nom de la ciutat amb l'índex parell original.

- a) Mostra el nou array omplert amb les funcions **print_r()** i **var_dump()**. Aquestes funcions són útils en el procés de depuració, ja que permeten imprimir el contingut de les variables.
- b) Crea una funció que li passis un array i retorni un string amb el següent format de llista:

<ul style="list-style-type: none">• CIUTAT 1 : Barcelona (0)• CIUTAT 2 : Lleida (2)• CIUTAT 3 : Badalona (4)• CIUTAT 4 : Sabadell (6)• CIUTAT 5 : Santa Coloma de Farners (8)
--

Mostra l'string resultant de passar-li a la funció el nou array omplert.

2. Treballarem amb el següent array associatiu de futbolistes i la respectiva quantitat de vegades que han guanyat la Pilota d'Or:

```
<?php
    $futbolistes = [
        "Franz Beckenbauer" => 2,
        "Johan Cruyff" => 3,
        "Alfredo Di Stéfano" => 2,
        "Lionel Messi" => 7,
        "Michel Platini" => 3,
        "Cristiano Ronaldo" => 5,
        "Marco van Basten" => 3
    ];
?>
```

Crea una funció per trobar i mostrar el futbolista que ha guanyat més vegades la pilota d'Or i quina quantitat n'ha guanyat.

Pots utilitzar un bucle amb foreach o la funció **max()** per trobar el valor més alt dins l'array i **array_search()** per trobar a quina clau correspon aquest valor.

3. Treballarem amb el següent array multidimensional on cada clau representa un ingredient i com a valor associat té un array que conté els plats que es poden fer amb l'ingredient:

```
<?php
$ingredientsPlats = [
    "Tomàquet" => ["Salsa de Tomàquet", "Gazpacho"],
    "Oli d'Oliva" => ["Hummus", "Pa amb oli", "Amanida Mediterrània"],
    "Formatge" => ["Pizza Margherita", "Lasagna", "Quiche Lorraine"],
    "All" => ["Sopa d'all", "Pa amb tomàquet"]
];
?>
```

Crea una funció que rebi un array multidimensional i generi i retorni un string formatat com una llista pels ingredients i una llista dins de cada llista d'ingredients per mostrar els plats, la llista ha d'estar precedida per un títol que també es rebrà com un paràmetre.

PLATS AGRUPATS PER INGREDIENTS

- Tomàquet
 - Salsa de Tomàquet
 - Gazpacho
- Oli d'Oliva
 - Hummus
 - Pa amb oli
 - Amanida Mediterrània
- Formatge
 - Pizza Margherita
 - Lasagna
 - Quiche Lorraine
- All
 - Sopa d'all
 - Pa amb tomàquet

4. Treballarem amb el següent array de números positius i negatius:

```
<?php
    $numerosPositiusNegatius = [1, -2, 3, -4, 5, 10, 9, -8, -7, 6];
?>
```

Crea una funció que recorri l'array i construeixi un nou array on, per cada element:

- Si el número és positiu, s'inserirà al final del nou array.
- Si el número és negatiu, s'inserirà al principi del nou array. Per afegir al principi d'un array pots usar la funció **array_unshift()**.

Per ajudar a entendre l'activitat, per l'array adjuntat, el resultat és el següent:

```
Array
(
    [0] => -7
    [1] => -8
    [2] => -4
    [3] => -2
    [4] => 1
    [5] => 3
    [6] => 5
    [7] => 10
    [8] => 9
    [9] => 6
)
```

D'aquest array resultant obté i mostra la següent informació:

- a) Suma de tots els números que conté.
- b) Números positius que conté.
- c) Quadrats dels números que conté.
- d) Ordena els números que conté de menor a major.
- e) Inverteix l'ordre dels elements que conté.
- f) Valor més petit que conté.
- g) Quantitat de números que conté.

Investiga les funcions necessàries.

BIBLIOGRAFIA I WEBGRAFIA

«PHP.net». <https://www.php.net/manual/es/index.php>

«w3Schools». <https://www.w3schools.com/php/>



Autor: Xavier Baubés Parramon

Aquest document es llicència sota Creative Commons versió 4.0.
Es permet compartir i adaptar el material però reconeixent-ne l'autor original.