



upcomillas *es*

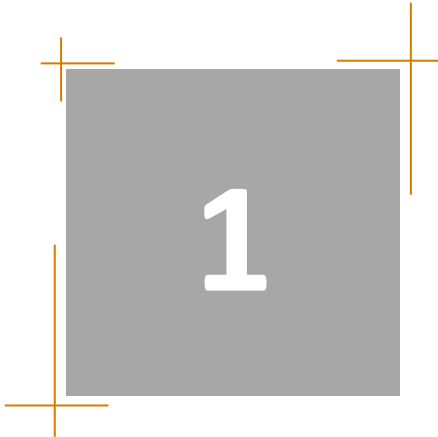
upcomillas *es*

5-PHP advanced concepts

Course: Developing web-based applications

Cristina Puente, Rafael Palacios

2010-2011



Arrays



Arrays

- A special variable, which can store multiple values in one single variable
- An array can hold all your variable values under a single name. And you can access the values by referring to the array name
- PHP supports both simple and multi-dimensional arrays
- In PHP, there are three kind of arrays
 - **Numeric Arrays:** An array with a numeric index
 - **Associative array:** An array where each ID key is associated with a value
 - **Multidimensional array:** An array containing one or more arrays

Numeric Arrays

- A numeric array stores each array element with a numeric index.
- There are two methods to create numeric arrays:
 - Index automatically assigned, as in the example:

```
$numbers=array("One", "Two", "Three", "Four");
```

- Index assigned manually, as in the example:

```
$numbers[0]="One";  
$numbers[1]="Two";  
$numbers[2]="Three";  
$numbers[3]="Four";
```

Associative Arrays

- Each ID key is associated with a value
- With associative arrays we can use the values as keys and assign values to them.

```
$ages = array  
("Rafael"=>32,  
"Cristina"=>28,  
"Juan"=>34);
```



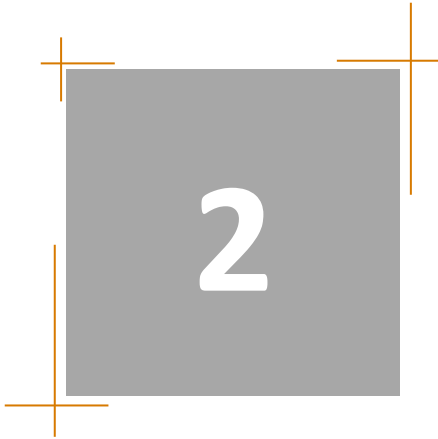
```
$ages['Rafael'] = "32" ;  
$ages['Cristina'] = "28";  
$ages['Juan'] = "34";
```

Multidimensional Arrays

- Each element in the main array can also be an array, and each element in the sub-array can be an array, and so on

```
$familias = array
(
    "Palacios"=>array
    (
        "Rafael",
        "Luis",
        "Marta"
    ),
    "Puente"=>array
    (
        "Cristina ",
        "Pilar ",
        "Manuel "
    )
);
```

```
echo "¿Es " . $familias['Puente'][2] .
    " parte de la familia Puente?";
```



Strings



Strings

- In PHP there are several functions and operators used to manipulate strings.
- `strlen(string)`. Used to know the length of a string.
- `split(separator,string)`. Divides a string using a character as separator.
- `sprintf(formatted string, var1, var2...)`. Writes a formatted string to a variable .

Strings

- `substr(string, start, length)`. Used to return a part of a string .
- `chop(string)`. Used to remove a white space or other predefined character from the right end of a string.
- `strpos(string1, string2)`. Returns the position of the first occurrence of a string inside another string (it is case-sensitive) .
- `str_replace(string1, string2, text)`. Replaces some characters in a string (case-sensitive) .

Strings

```
<?php
    echo strlen("12345"),"<br>";

    $palabras=split(" ","Esto es una prueba");
    for($i=0;$palabras[$i];$i++)
    echo $palabras[$i],"<br>";

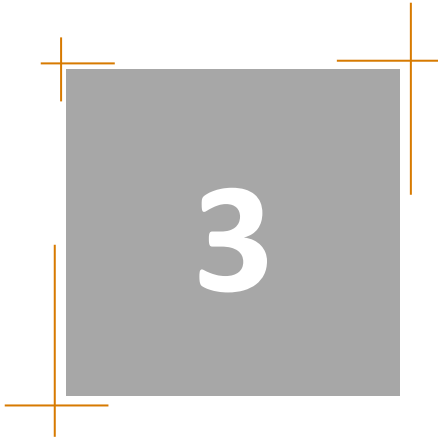
    $resultado=sprintf("8x5 = %d <br>",$8*5);
    echo $resultado,"<br>";

    echo substr("Devuelve una subcadena de otra",9,3),"<br><br>";

    if (chop("Cadena \n\n ") == "Cadena")
        echo "Iguales<br><br>";

    echo strpos("Busca la palabra dentro de la frase", "palabra"),"<br><br>";

    echo str_replace("verde","rojo","Un pez de color verde, como verde es la
    hierba."),"<br>";
?>
```



Functions



Functions

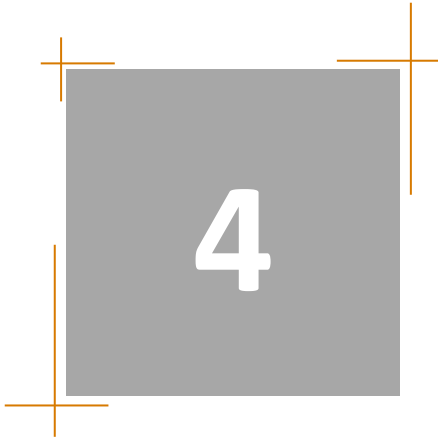
- Used to gather a set of instructions. You may call a function from anywhere within a page.

```
<?php
    function Nombre(parametro1, parametro2...)
    {
        instrucción1;
        instrucción2;
        instrucción3;
        instrucción4;
        return value;
    }
?>
```

Functions

- To add more functionality to a function, we can add parameters.
- Parameters are specified after the function name, inside the parentheses

```
<?php  
  
function mediaaritmetica($a, $b)  
{  
    $media=($a+$b)/2;  
    return $media;  
}  
echo media_aritmetica(4,6),"<br>";  
echo media_aritmetica(3242,524543),"<br>";  
?>
```



Forms



Form Handling

- Any form element in an HTML page will **automatically** be available to your PHP scripts

```
<html>
  <body>
    <form action= "bienvenido.php" method="post">
      Name: <input type="text" name= "nombre" />
      Age: <input type="text" name= "edad" />
      <input type="submit" />
    </form>
  </body>
</html>
```

```
<html>
  <body>
    Bienvenido <?php echo $_POST["nombre"]; ?>!  

    Tienes<?php echo $_POST["edad"]; ?> años!!
  </body>
</html>
```

Form Handling

- The built-in `$_GET` function is used to collect values from a form sent with `method="get"`
- Information sent from a form with the GET method is visible to everyone

```
<body>
  Bienvenido <?php echo $_GET["nombre"]; ?>!<br />
  Tienes<?php echo $_GET["edad"]; ?> años!!.
</body>
```

- Information sent from a form with the POST method is invisible to others and has no limits on the amount of information to send

```
<body>
  Bienvenido <?php echo $_POST["nombre"]; ?>!<br />
  Tienes<?php echo $_POST["edad"]; ?> años!!.
</body>
```




Escuela Técnica Superior de Ingeniería ICAI

Alberto Aguilera 25
28015 Madrid
Tel +34 91 542 28 00
Fax + 34 91 542 31 76
lwww.icaí.upcomillas.es

www.upcomillas.es

