

Tema: Gestión de Cadenas

Contenido

En esta practica se abordan una seria de funciones PHP para gestión de cadenas que facilitan el uso y transformación de las mismas, entre las operaciones de gestión de cadenas que se pueden realizar se puede mencionar, mostrar una cadena por la salida estándar, formatear una cadena, substraer una subcadena de una cadena, entre otras operaciones.

Objetivo Especifico

1. Utilizar funciones de gestion de cadenas mas ampliamente utilizadas en script en PHP.
2. Crear scripts en PHP para solucionar problemas tipicos de gestion de datos proporcionados por el usuario tales como validarlos, generacion de informes en html, entre otros.

Material y Equipo

1. Editor de Texto
2. Navegador Web Firefox
3. WampServer

Introduccion Teorica

Gestion de cadenas en PHP

PHP dispone de un conjunto de funciones de gestion de cadenas que facilitan el uso y transformación de las mismas, entre estas funciones se pueden mencionar las siguientes:

- a) print, echo, printf
- b) ltrim, rtrim, trim
- c) strtolower, strtoupper
- d) str_replace, strstr, substr_coun, strpos, strrpos
- e) Entre otras funciones de importancia.

Procedimiento

Utilizando su editor de texto favorito, digitar el código de cada uno de los ejemplos indicados, **código de las tablas 1 a 6**, posteriormente ejecutar el código proporcionado de cada ejemplo utilizando un navegador web.

cadenas01.php

```
<?php
$n = 43951789;
$u = -43951789;
$c = 65; // ASCII 65 es 'A'

// fijarse en el doble %, esto muestra un caracter '%' literal
printf("%%b = '%b'\n", $n); // representación binaria
printf("%%c = '%c'\n", $c); // muestra el caracter ascii, igual que la función chr()
printf("%%d = '%d'\n", $n); // representación estándar de un entero
printf("%%e = '%e'\n", $n); // notación científica
printf("%%u = '%u'\n", $n); // representación sin signo de un entero positivo
printf("%%u = '%u'\n", $u); // representación sin signo de un entero negativo
printf("%%f = '%f'\n", $n); // representación de punto flotante
printf("%%o = '%o'\n", $n); // representación octal
printf("%%s = '%s'\n", $n); // representación en una cadena
printf("%%x = '%x'\n", $n); // representación hexadecimal (minúsculas)
printf("%%X = '%X'\n", $n); // representación hexadecimal (mayúsculas)

printf("%%+d = '%+d'\n", $n); // especificador de signo sobre un entero positivo
printf("%%+d = '%+d'\n", $u); // especificador de signo sobre un entero negativo
?>
```

Tabla 1

cadenas02.php

```
<?php
print("Hola estudiantes PHP<br>");

print "print() también funciona sin paréntesis.<br>";

print "Esto separa
múltiples líneas. Los saltos de línea también
se mostrarán<br>";

print "Esto separa\nmúltiples líneas. Los saltos de línea también\nse
mostrarán.<br>";

print "Para escapar caracteres se \"hace así\".<br>";

// También se puede usar variables usando print
$var01 = "Universidad<br>";
$var02 = "Don Bosco<br>";
```

```
print "var01 es $var01<br>";

// También se pueden usar arrays
$arreglo01 = array("carnet" => "SC0003");

print "arreglo01 es {$arreglo01['carnet']} <br>"; //

// Al usar comillas simples se mostrará el nombre de la variable, no su valor
print 'var01 is $var01<br>';

// Si no se necesita mostrar otros caracteres, se puede simplemente mostrar el
contenido de las variables
print $var01;

?>
```

Tabla 2**cadenas03.php**

```
<?php
$s = 'planetas';
$t = 'muchos planetas';

printf("[%s]<br>", $s); // salida estándar de string
printf("[%10s]<br>", $s); // justificación a la derecha con espacios
printf("[%10s]<br>", $s); // justificación a la izquierda con espacios
printf("[%010s]<br>", $s); // relleno con ceros también funciona sobre strings
printf("[%10s]<br>", $s); // utiliza el caracter de relleno personalizado '#'
printf("[%10.10s]", $t); // justificación a la izquierda pero con un corte a los 10
caracteres

?>
```

Tabla 3

cadenas04.php

```

<html>
<head>
  <title>Gestion de Cadenas</title>
</head>
<body>
  <center>
    <h2>Función <i>str_pad</i></h2>
    <?php
      $cadena="0123456789";
      $long = 20;
      $car = ".";
      echo "<table border='1' cellpadding='2' cellspacing='2'>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>cadena</td><td>$cadena</td>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>str_pad(cadena,$long,'$car')</td>";
      echo "<td>".str_pad($cadena,$long,$car)."</td></tr>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>str_pad(cadena,
                                $long,'$car',STR_PAD_RIGHT)</td>";
      echo "<td>".str_pad($cadena,$long,
                                $car,STR_PAD_RIGHT)."</td></tr>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>str_pad(cadena,
                                $long,'$car',STR_PAD_LEFT)</td>";
      echo "<td>".str_pad($cadena,$long,$car,STR_PAD_LEFT)."</td></tr>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>str_pad(cadena,
                                $long,'$car',STR_PAD_BOTH)</td>";
      echo "<td>".str_pad($cadena,$long,
                                $car,STR_PAD_BOTH)."</td></tr>\n";
      echo "</table>\n";
    ?>
  </center>
</body>
</html>

```

Tabla 4

cadenas05.php

```

<html>
<head>
  <title>Gestion de Cadenas</title>
</head>
<body>
  <center>
    <h2>Funciones <i>strcmp y strcasecmp</i></h2>
    <?php
      $cad1="atención";
      $cad2="atención";
      echo "<table border='1' cellpadding='2' cellspacing='2'>\n";
      echo "<tr align='center'><td bgcolor='cyan'>cadena1</td>";
      echo "<td>$cad1</td></tr>\n";
      echo "<tr align='center'><td bgcolor='cyan'>cadena2</td>";
      echo "<td>$cad2</td></tr>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>strcmp(cadena1,cadena2)</td>";
      echo "<td>".strcmp($cad1,$cad2)."</td></tr>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>strcasecmp(cadena1,cadena2)</td>";
      echo "<td>".strcasecmp($cad1,$cad2)."</td></tr>\n";
      echo "</table><br>\n";

      $cad1="Atención";
      $cad2="atención";
      echo "<table border='1' cellpadding='2' cellspacing='2'>\n";
      echo "<tr align='center'><td bgcolor='cyan'>cadena1</td>";
      echo "<td>$cad1</td></tr>\n";
      echo "<tr align='center'><td bgcolor='cyan'>cadena2</td>";
      echo "<td>$cad2</td></tr>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>strcmp(cadena1,cadena2)</td>";
      echo "<td>".strcmp($cad1,$cad2)."</td></tr>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>strcasecmp(cadena1,cadena2)</td>";
      echo "<td>".strcasecmp($cad1,$cad2)."</td></tr>\n";
      echo "</table>\n";

    ?>
  </center>
</body>
</html>

```

Tabla 5

cadenas06.php

```

<html>
<head>
  <title>Gestion de Cadenas</title>
</head>
<body>
  <center>
    <h2>Funciones <i>explode e implode</i></h2>
    <?php
      $patron=" ";
      $nexo=":";
      $limite=5;
      $cadena="dato1 dato2 dato3 dato4 dato5";
      $datos = explode($patron,$cadena,$limite);
      echo "<table border='1' cellpadding='2' cellspacing='2'>\n";
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>cadena</td><td>$cadena</td>\n";
      echo "<tr align='center'>";
      echo "<td colspan='2' bgcolor='cyan'>explode('$patron',cadena,$limite)</td>";
      echo "</tr>\n";
      for($i=0;$i<$limite;$i++){
        echo "<tr align='center'>";
        echo "<td bgcolor='cyan'>subcadena $i</td>";
        echo "<td>$datos[$i]</td></tr>\n";
      }
      echo "<tr align='center'>";
      echo "<td bgcolor='cyan'>implode('$nexo',datos)</td>";
      echo "<td>\",implode($nexo,$datos),\"</td></tr>\n";
      echo "</tr>\n";
      echo "</table>\n";
    ?>
  </center>
</body>
</html>

```

Tabla 6

Utilizando su editor de texto favorito, digitar el código del script php **validar_correo.php**, posteriormente ejecutar el código proporcionado utilizando un navegador web.

validar_correo.php

```
<html>
<head>
  <title>Validación de direcciones de correo</title>
</head>
<body>
<center>
<h2>Validacion de direcciones de correo</h2>
<form action="validar_correo.php" method="post">
<?php
  echo '<p><input type="text" name="cadena" value="" size=50>';
?>
  <p><input type="submit" value="validar">
    <input type="reset" value="limpiar">
</form>
<?php
// Función que comprueba que no haya dos puntos seguidos
$cadena = $_POST['cadena'];
function comprobar_puntos($cadena)
{
  $seguir = true;
  // Se obtiene la posición del primer punto
  $posicion = strpos($cadena, '.');
  // Se comprueba que exista
  while (($posicion) && ($seguir))
  {
    // Se comprueba que el anterior y el siguiente no son . o @
    if (($cadena[$posicion - 1] != '.') &&
        ($cadena[$posicion + 1] != '.') &&
        ($cadena[$posicion - 1] != '@') &&
        ($cadena[$posicion + 1] != '@'))
      // Se obtiene la subcadena a partir del punto
      $cadena = substr($cadena, $posicion + 1);
    else
      $seguir = false;
    // Se vuelve a obtener la posición del primer punto
    $posicion = strpos($cadena, '.');
  }
  return $seguir;
}

function validar_correo ($cadena)
{
  // Primero se comprueba que sólo aparezca una arroba
  // y que al menos exista un punto decimal. Para lo que
  // se calcula la frecuencia de los caracteres
  $caracteres = count_chars($cadena);
  if (($caracteres[ord('@')] == 1) && ($caracteres[ord('.')]>0))
  {
```

```

// Se comprueba que sólo existan caracteres alfabéticos
$caracteres = count_chars($cadena, 3);
$i = 0;
$seguir = true;
while (($i < strlen($cadena)) && $seguir)
{
    $posicion = ord($cadena[$i]);
    if ($posicion < ord('0'))
    {
        // El código del carácter no es numérico
        if ($posicion != ord('.')) // y no es el punto
            $seguir = false;
    }
    elseif (($posicion > ord('9')) && ($posicion < ord('@')))
        $seguir = false; // Entre números y letras mayúsculas
    elseif (($posicion > ord('Z')) && ($posicion < ord('a')))
        $seguir = false; // Entre mayúsculas y minúsculas
    elseif ($posicion > ord('z'))
        $seguir = false; // mayor que letras minúsculas
    $i++;
}
if ($seguir)
{
    // comprobar que antes de @ hay caracteres
    // y al menos un punto
    $posicion1 = strpos($cadena, '@');
    $posicion2 = strrpos($cadena, '.');
    $seguir = (($posicion1 + 1 <= $posicion2) &&
        ($posicion2 < (strlen($cadena) - 1)) &&
        ($posicion1 <> 0));
    if ($seguir)
        $seguir = comprobar_puntos($cadena);
}

return $seguir;
}
else
{
    return false;
}
}

// Determina si hay o no cadena de caracteres
if (!empty($cadena))
    if (validar_correo($cadena))
        echo "'$cadena' es una dirección de correo válida<br>";
    else
        Echo "'$cadena' no es una dirección de correo válida<br>";
?>
</center>
</body>
</html>

```

Tabla 7

Guía 7: Gestión de Cadenas

Alumno:

Maquina No:

Docente:

GL:

Fecha:

EVALUACION					
	%	1-4	5-7	8-10	Nota
CONOCIMIENTO	Del 20 al 30%	Conocimiento deficiente de los fundamentos teóricos	Conocimiento y explicación incompleta de los fundamentos teóricos	Conocimiento completo y explicación clara de los fundamentos teóricos	
APLICACIÓN DEL CONOCIMIENTO	Del 40% al 60%				
ACTITUD					
	Del 15% al 30%	No tiene actitud proactiva.	Actitud propositiva y con propuestas no aplicables al contenido de la guía.	Tiene actitud proactiva y sus propuestas son concretas.	
TOTAL	100 %				