

Guia PHP

Pedro Monreal Sánchez

- [Guia PHP](#)
 - [Pedro Monreal Sánchez](#)
 - [Variables](#)
 - [Intval](#)
 - [Strval](#)
 - [Condicionales](#)
 - [operador ternario](#)
 - [is_int](#)
 - [Tratar strings](#)
 - [strlen](#)
 - [substr](#)
 - [strpos](#)
 - [Trim](#)
 - [strtoupper](#)
 - [strtolower](#)
 - [str_replace](#)
 - [range](#)
 - [Funciones](#)
 - [Variable opcional en funciones](#)
 - [Arrays](#)
 - [is_array](#)
 - [print_r \(\\$array\);](#)
 - [array_key_exists](#)
 - [array_values \(Elimina huecos vacíos\)](#)
 - [explode](#)
 - [explode con saltos de linia \n](#)
 - [implode](#)
 - [in_array](#)
 - [isset](#)
 - [array_search](#)
 - [sort](#)
 - [asort](#)
 - [ksort](#)
 - [krsort](#)
 - [shuffle](#)
 - [foreach](#)
 - [Recorrer un array asociativo](#)
 - [empty\(\)](#)
 - [Count](#)
 - [matriz](#)
 - [Array](#)

- array_merge
- array asociativo
- array_column
- Bucles
 - for
 - For de 3 en 3
 - While
 - do while
 - tabla con html
 - tabla con html array asociativo
- Operadores
 - \$val+= 5;
- HTML + PHP
 - Pasar info por URL
 - **Form** + **post** en el mismo fichero
 - Tabla con arrays formulario
 - Radio button sticky array asociativo
 - Select array asociativo
- Tablas
 - Tabla con array asociativo básico
- Cookies
 - Establecer cookie
 - Obtener cookie
 - Eliminar cookie
- Base de datos
 - PDO connect
 - PDO file
 - Consulta SQL
 - Insert SQL
 - Delete SQL
 - Pasar info por URL
 - Ejemplo completo SQL

Variables

Intval

Convierte un string a un entero.

Strval

Convierte un entero a un string.

Condicionales

operador ternario

Un if comprimido

```
condición ? valor_si_verdadero : valor_si_falso;
```

is_int

Nos dice si es un int o no

is_int(\$value)

```
$value = 2;
if(is_int($value)){
    echo "INT";
}
else
{
    echo "NO INT";
}
echo "<br>";
```

RESULTADO

INT

Tratar strings

strlen

Dice la longitud en caracteres de una variable

```
$var1="hola";
echo strlen($var1);
```

RESULTADO

4

substr

Extraer de una cadena la cantidad que queremos

```
echo substr('abcdef', 3, 1);
```

RESULTADO

```
d
```

strpos

La función `strpos()` en PHP se utiliza para encontrar la posición de la primera aparición de una subcadena dentro de una cadena. Devuelve el índice de la primera ocurrencia de la subcadena (empezando desde 0) o `false` si la subcadena no se encuentra.

```
strpos("cadena_completa", "subcadena_a_buscar");

$cadena = "Hola, ¿cómo estás?";
$posicion = strpos($cadena, "cómo");

if ($posicion !== false) {
    echo "La palabra 'cómo' se encuentra en la posición: " . $posicion;
} else {
    echo "La palabra 'cómo' no se encuentra en la cadena.";
}
```

RESULTADO

```
La palabra 'cómo' se encuentra en la posición: 6
```

Trim

Sirve para eliminar espacios en blanco u otros caracteres específicos.

```
$texto = "  Hola mundo  ";
$resultado = trim($texto);
echo $resultado; // Salida: "Hola mundo"
```

strtoupper

Pasa un string a mayúsculas

```
$userid = "PedroMonreal123";
$userid = strtoupper($userid);
echo $userid; // Salida: "PEDROMONREAL123"
```

strtolower

Pasa un string a minusculas

```
$userid = strtolower($userid);
```

str_replace

Reemplaza caracteres

```
$date="12/12/2012";  
$date=str_replace("/", "-", $date);  
  
//Otra forma, usando un array de formas  
$date=str_replace(["*", ";"], "-", $date);  
  
echo $date."<br>\n";
```

range

Genera un array que contiene los caracteres o numeros en el rango.

```
<?php  
// Crear un array con los números del 0 al 9  
$digits = range(0, 9);  
  
// Crear un array con las letras de la A a la F  
$digits2 = range("A", "F");  
  
// Mostrar el contenido de $digits  
echo "Array de dígitos del 0 al 9:\n";  
print_r($digits);  
  
// Mostrar el contenido de $digits2  
echo "Array de letras de la A a la F:\n";  
print_r($digits2);  
?>
```

Funciones

Variable opcional en funciones

\$language = 'en'

Por defecto, si no se pasa el valor, tomará el valor **"en"**

```
function getWeekDayStr($number, $language = 'en'){  
  
}
```

Arrays

is_array

Dice si el valor es un array o no, devuelve true o false.

```
$array=["a", "s", "g"];  
  
var_dump(is_array($array));
```

```
print_r ($array);
```

Muestra el contenido de un array

array_key_exists

Verifica si el valor está en el array asociativo

"También se puede usar isset"

```
if (array_key_exists($language, $days)) {  
  
}
```

array_values (Elimina huecos vacíos)

Reindexa los valores de array

Elimina los huecos vacíos

```
$names=["Pedro","Christian","Biel"]  
  
$names = array_values($names);
```

explode

Convierte string en array, estando separado por un carácter, por ejemplo ;

```
function showArray($array){
    echo "<pre>";
    print_r($array);
    echo "</pre>";
}

$dataStr="Pedro;Monreal;717776040;2004";

$dataArray=explode(";", $dataStr);
showArray($dataArray);
```

Resultado

```
(
    [0] => Pedro
    [1] => Monreal
    [2] => 717776040
    [3] => 2004
)
```

explode con saltos de linea \n

```
<?php

$string="Surname;Name;Address;Phone
Rigo;Andreu;C/ My Street, 99;666999666
Doe;John;C/ 5th Avenue, 3;333111333
";

$newarray=explode("\n",$string);
var_dump($newarray);
?>
```

implode

Convierte array en string, estando separado por un carácter, por ejemplo ;

```
$dataArray=["Pedro","Monreal","717776040","2004"];

$Valores=implode(" ", $dataArray);

echo $Valores;
```

Resultado

```
Pedro,Monreal,717776040,2004
```

in_array

Ver si existe un valor dentro de un array

```
$colores=["rojo", "verde", "azul"];  
var_dump(in_array("verde",$colores));
```

Resultado

```
bool(true)
```

isset

En PHP, `isset()` es una función que determina si una variable está definida y no es null. Devuelve `true` si la variable existe y tiene un valor distinto de null, y `false` en caso contrario. Es muy útil para verificar si una variable fue inicializada antes de intentar usarla, evitando errores.

```
var_dump(isset($phoneNumber))
```

array_search

Sirve para buscar un valor asociado a otro

```
$phoneNumbers = [  
    "John" => "123-456-789",  
    "Jane" => "555-555-555",  
    "Mike" => "666-666-666",  
    "Sara" => "777-777-777"  
];  
  
$testIndex = array_search("666-666-666", $phoneNumbers);  
var_dump($testIndex);
```

Resultado

```
string(4) "Mike"
```

sort

Ordena valores de array tipo index

```
function showArray($array){  
    echo "<pre>";  
    print_r($array);  
    echo "</pre>";  
}  
  
$colors = array("rojo", "azul", "verde", "amarillo"); // Definir el array  
  
$colorsv2 = $colors;  
showArray($colorsv2);  
  
sort($colors);  
showArray($colors);
```

RESULTADO

```
Array  
(  
    [0] => rojo  
    [1] => azul  
    [2] => verde  
    [3] => amarillo  
)  
Array  
(  
    [0] => amarillo  
    [1] => azul  
    [2] => rojo  
    [3] => verde  
)
```

asort

Ordena un array asociativo

```
asort($phoneNumbers);
```

ksort

Ordena un array asociativo por clave

```
ksort($phoneNumbers);
```

krsort

Ordena un array asociativo por clave al revés

```
krsort($phoneNumbers); //reverse order
```

shuffle

Desordenar los valores del array

```
function showArray($array){  
    echo "<pre>";  
    print_r($array);  
    echo "</pre>";  
}  
  
$colors = array("rojo", "azul", "verde", "amarillo"); // Definir el array  
  
$colorsv2 = $colors;  
showArray($colorsv2);  
  
shuffle($colors);  
showArray($colors);
```

RESULTADO

```
function showArray($array){  
    echo "<pre>";  
    print_r($array);  
    echo "</pre>";  
}  
  
$colors = array("rojo", "azul", "verde", "amarillo"); // Definir el array  
  
$colorsv2 = $colors;  
showArray($colorsv2);  
  
shuffle($colors);  
showArray($colors);
```

foreach

Recorrer un array

```
$colors = array("rojo", "azul", "verde", "amarillo");
echo "<ol>\n";
foreach ($colors as $color) {
    echo "<li>$color</li>\n";
}
echo "</ol>\n";
```

RESULTADO

```
rojo
azul
verde
amarillo
```

Recorrer un array asociativo

```
$phoneNumbers = array(
    "Casa" => "123-456-7890",
    "Trabajo" => "098-765-4321",
    "Móvil" => "555-123-4567"
);

echo "<ul>\n";
foreach ($phoneNumbers as $key => $value) {
    echo "<li>$key: $value</li>\n";
}
echo "</ul>\n";
```

RESULTADO

```
Casa: 123-456-7890
Trabajo: 098-765-4321
Móvil: 555-123-4567
```

Añade el campo Mike con sus atributos.

```
$users = [ "John" => ["age" => 25, "city" => "New York"], "Jane" => ["age" =>
22, "city" => "Paris"] ];
$users["Mike"] = ["age" => 30, "city" => "Berlin"];
showArray($users);
```

empty()

Verificar si un array está vacío.

```
if (empty($values)){  
    return false;  
}
```

Count

Dice la longitud del array

```
count($array);
```

matriz

```
$matrix = array(  
    array(1, 2, 3),  
    array(4, 5, 6),  
    array(7, 8, 9)  
);
```

Array

La función array crea un array con los valores establecidos.

```
$array1 = array(1, 2, 3);
```

array_merge

```
$mergedArray = array_merge($digits, $letters);
```

array asociativo

```
$phoneNumbers = array(  
    "Pedro" => "123-456-7890",  
    "Ana" => "987-654-3210",  
    "Luis" => "555-555-5555",  
    "Marta" => "333-333-3333"
```

```
);
```

array_column

Extrae una columna de un array asociativo

```
array(2) { [0]=> array(1) { ["weight"]=> float(15) } [1]=> array(1) { ["weight"]=> float(90) } }
```

```
$weights = array_column($rows, 'weight');
```

Bucles

for

```
echo "<h2>Tabla del 5</h2>";  
for($i=1; $i<=10;$i++){  
echo $i." x ". 5 . " = ". ($i*5) . "<br>\n";  
}
```

RESULTADO

```
Tabla del 5  
1 x 5= 5  
2 x 5= 10  
3 x 5= 15  
4 x 5= 20  
5 x 5= 25  
6 x 5= 30  
7 x 5= 35  
8 x 5= 40  
9 x 5= 45  
10 x 5= 50
```

For de 3 en 3

```
echo "<h2>De 3 en 3</h2>";  
for($i=30; $i<=50;$i+=3){  
echo $i . ", ";  
}
```

While

```
<?php

$a=8;
$i=1;

while ($i<=10){
    echo "$a x $i = " .($a*$i). "<br>";
    $i++;
}
?>
```

RESULTADO

```
8 x 1 = 8
8 x 2 = 16
8 x 3 = 24
8 x 4 = 32
8 x 5 = 40
8 x 6 = 48
8 x 7 = 56
8 x 8 = 64
8 x 9 = 72
8 x 10 = 80
```

do while

```
<?php
$a = 8;
$i = 1;

do {
    echo "$a x $i = " . ($a * $i) . "<br>";
    $i++;
} while ($i <= 10);

?>
```

RESULTADO

```
8 x 1 = 8
8 x 2 = 16
8 x 3 = 24
```

```

8 x 4 = 32
8 x 5 = 40
8 x 6 = 48
8 x 7 = 56
8 x 8 = 64
8 x 9 = 72
8 x 10 = 80

```

tabla con html

```

<?php

echo "<table border=1>\n";
for ($r=1; $r<=10;$r++){
    echo "<tr>";
    for($c=1;$c<=10;$c++){
        echo "<td>".($r*$c)."</td>";
    }
    echo "</tr>\n";
}
echo "</table>\n";

?>

```

RESULTADO

1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

tabla con html array asociativo

```

<?php

function array2table($col1, $col2, $array) {
    $table="<table border=solid>\n";
    if($col1 != ("") && $col2 !=("")){
        $table.="<tr><th>$col1</th><th>$col2</th></tr>\n";
    }
    foreach ($array as $v1 => $v2) {

```

```
        $table.= "<tr><td>$v1</td><td>$v2</td></tr>\n";
    }
    $table.="</table>\n";
    return $table;
}

$datos = [
    "Pedro" => "848559340",
    "Miguel" => "384442023",
    "Juan" => "444556732"
];

echo array2table("Name", "Phone", $datos);

?>
```

RESULTADO

Name	Phone
Pedro	848559340
Miguel	384442023
Juan	444556732

Operadores

`$val+= 5;`

Sirve para sumer un valor a una variable funciona igual que:

```
$val = $val + 5;
```

HTML + PHP

Pasar info por URL

Podmeos pasar información de una web a otra, se debe emplear:

Origen

```
href='index.php?nombre='Pedro'
```

Destino


```
$_GET['nombre'];
```

Form + post en el mismo fichero

```
<!doctype html>
<html lang="en">

<head>
    <!-- change to utf-8 if you use that -->
    <meta charset="iso-8859-1">
    <title>BMI Calculator</title>
</head>
<body>
    <h1>BMI Calculator</h1>
    <form action="" method="post">
        <label for="weight">Weight (Kg)</label>
        <input type="number" name="weight">
        <label for="height">Height (m)</label>
        <input type="number" name="height">
        <input type="submit">
    </form>

    <?php
    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        if (!isset($_POST["weight"], $_POST["height"]) || empty($_POST["weight"]) ||
empty($_POST["height"])) {
            echo "Empty values";
        }

        else
        {
            echo "Your weight is " . $_POST["weight"]. " Kg and your height is " .
$_POST["height"]. " m <br><br>";
            echo BMI($_POST["weight"], $_POST["height"]);
        }

        }
        function BMI($weight, $height){

            $IMC = $weight/$height**2;

            if ($IMC < 18.5){
                return "Low weight";
            }
            elseif ($IMC > 18.5 && $IMC < 24.9 ){

                return "Normal weight";
            }

        }
```

```
elseif ($IMC > 24.9 && $IMC < 29.9)
{
    return "Overweight";
}
elseif ($IMC > 30){
    return "Obesity";
}
}

?>

</body>
</html>
```

PARTE IF

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {}
```

En el caso del formulario, podemos usar if GET para que solo muestre el formulario al cargar la página, si se hace el POST no se mostrará.

Tabla con arrays formulario

```
<!doctype html>
<html lang="en">

<head>
    <meta charset="iso-8859-1">
    <title>Inverting Array Table</title>
</head>

<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if(isset($_POST["values"]) && $_POST["values"]!=""){
        $NewArray=explode(",",$_POST["values"]);
        echo "The list introduced is: <br>";
        echo array2row($NewArray);
        echo "The reversed list is:<br>";
        $array2rowValues=array2row(invertarray($NewArray));
        if($array2rowValues==FALSE){
            echo "ERROR";
        }
        else
            echo $array2rowValues;
    }
    else
    {
        echo "ERROR";
    }
}
```

```

    }
}

function invertarray($array){
    if(isset($array) && $array!=""){
        for($i=(count($array)-1); $i >= 0; $i--){
            $arrayInvertido[]=$array[$i];
        }
        return($arrayInvertido);
    }
    else
    {
        return FALSE;
    }
}

function array2row($array){
    if(isset($array)){
        $html = "<table>";
        foreach($array as $valor){
            $html .= "<th>$valor</th>";
        }
        $html .= "</table>";
        return $html;
    }
    else
    return FALSE;
}
?>

<body>
    <form action="" method="post">
        <h1>Introduce a list separating by commas</h1>
        <textarea name="values"></textarea>
        <input type="submit">
    </form>
</body>

</html>

```

Radio button sticky array asociativo

```

<!doctype html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <title></title>

```

```
</head>
<?php

$array = [
    "Toy_Story" => "First Pixar film",
    "The_incredible" => "Better than Marvel",
    "Cars" => "Best film about cars, ever",
    "WALL-E" => "Is it our future?",
    "Lightyear" => "The last one, until now"
];
?>
<body>
    <form action="" method="post">
        <p>Choose one from the following ones by Pixar.</p>
        <?php
        foreach ($array as $film => $description) {
            if ($_SERVER["REQUEST_METHOD"] == "POST") {
                if ($_POST["radiobutton"] == $film) {
                    echo "<input type='radio' id=$film name='radiobutton'
value=$film checked>";
                    echo "<label for=$film>$film</label><br>";
                } else {
                    echo "<input type='radio' id=$film name='radiobutton'
value=$film>";
                    echo "<label for=$film>$film</label><br>";
                }
            } else {
                echo "<input type='radio' id=$film name='radiobutton'
value=$film>";
                echo "<label for=$film>$film</label><br>";
            }
        }
        ?>
        <br>
        <input type="submit">
    </form>

    <?php
    if ($_SERVER["REQUEST_METHOD"] == "POST") {

        echo $_POST["radiobutton"] . ": ". $array[$_POST["radiobutton"]];

    }

    ?>
</body>

</html>
```

Select array asociativo

```

<!doctype html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <title></title>
</head>
<?php
$array=[
    "6.06"=>["Code-Name"=>"Dapper Drake","Relase-Date"=>"2006-06-01"],
    "8.04"=>["Code-Name"=>"Hardy Heron","Relase-Date"=>"2008-04-24"],
    "10.04"=>["Code-Name"=>"Lucid Lynx","Relase-Date"=>"2010-04-29"],
    "12.04"=>["Code-Name"=>"Precise Pangolin","Relase-Date"=>"2012-04-26"]
]
?>
<body>
    <form action="" method="post">
        <label for="version">Version</label>
        <select name="version" id="version">
            <option value="6.06">6.06</option>
            <option value="8.04">8.04</option>
            <option value="10.04">10.04</option>
            <option value="12.04">12.04</option>
        </select>
        <br><br>
        <label for="property">Property</label>
        <select name="property" id="property">
            <option value="Code-Name">Code-Name</option>
            <option value="Relase-Date">Relase-Date</option>
        </select>
        <input type="submit">
    </form>
    <?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    echo $array[$_POST["version"]][$_POST["property"]];
}

?>
</body>

</html>

```

Tablas

```

<table>
    <thead> <!-- Encabezado de la tabla -->
        <tr> <!-- Fila del encabezado -->
            <th>Columna 1</th>
            <th>Columna 2</th>

```

```
        <th>Columna 3</th>
    </tr>
</thead>
<tbody> <!-- Cuerpo de la tabla -->
    <tr> <!-- Primera fila de datos -->
        <td>Dato 1</td>
        <td>Dato 2</td>
        <td>Dato 3</td>
    </tr>
    <tr> <!-- Segunda fila de datos -->
        <td>Dato 4</td>
        <td>Dato 5</td>
        <td>Dato 6</td>
    </tr>
</tbody>
</table>
```

Tabla con array asociativo básico

```
<!doctype html>
<html lang="en">

<head>
    <meta charset="iso-8859-1">
</head>
<?php
$string = [
    ["surname" => "Rigo", "name" => "Andreu", "address" => "My Street, 99",
    "phone" => "666999666"],
    ["surname" => "Doe", "name" => "John", "address" => "5th Avenue, 3", "phone"
=> "333111333"]
];

echo "<table >";
echo "<td>Surname</td>";
echo "<td>Name</td>";
echo "<td>Address</td>";
echo "<td>Phone</td>";

foreach ($string as $case) {
    echo "<tr>";
    foreach ($case as $value) {
        echo "<td>$value</td>";
    }
    echo "</tr>";
}
?>

<body>
```

```
</body>
<style>
    table,
    th,
    td {
        border: 1px solid black;
    }
</style>

</html>
```

Cookies

Establecer cookie

```
setcookie("number2guess","Valor cookie",(time()+3600*24*30));
```

Obtener cookie

```
$_COOKIE["number2guess"]
```

Eliminar cookie

Se emplea isset para verificar que exista.

```
if (isset($_COOKIE['number2guess'])) {
    setcookie('number2guess','', (time()-3600));
}
```

Base de datos

PDO connect

```
require('./pdo_connect.php');
```

PDO file

```
<?php
$flag=FALSE;
try {
```

```
$pdo = new PDO('mysql:host=localhost;dbname=register;charset=utf8mb4',
'databasePHP', '');
$output = 'Database connection established.';
} catch (PDOException $e) {
    $output = 'Unable to connect to the database server: ' . $e->getMessage(). '
in ' . $e->getFile() . ':' . $e->getLine();
    $flag=TRUE;
}
if ($flag) die("<p>$output</p>");
else echo "<p>$output</p>";
?>
```

Consulta SQL

```
$q = "SELECT last_name, first_name, DATE_FORMAT(registration_date, '%M %d, %Y') AS
dr, user_id FROM users ORDER BY registration_date ASC";

$result = $pdo->query($q);
$rows=$result->fetchAll();
```

Insert SQL

```
$nombre = "HOLA";

$queryOK=TRUE;
try {
    $q = "INSERT INTO Nombres (nombre) VALUES (:nombre)";

    $stmt = $pdo->prepare($q);
    $stmt->bindValue(':nombre',$nombre);
    $stmt->execute();
} catch(PDOException $ex) {
    $queryOK=FALSE;
    $output = 'Database error: ' . $ex->getMessage() . ' in ' . $ex->getFile() .
    ':' . $ex->getLine();
}
if ($queryOK) {
    echo "Register added to Database<br>";
} else {
    echo "Please try again later<br>";
    echo "Debug information:<br>";
    echo $output."<br>";
    echo $q."<br>";
}
```

Delete SQL


```
<?php
    require('pdo-connect.php');

    $queryOK = TRUE;

    try {
        $q = "DELETE FROM trabajadores WHERE dni = :dni";

        $stmt = $pdo->prepare($q);
        $stmt->bindValue(":dni", $dni);
        $stmt->execute();

    } catch (PDOException $ex){

        $queryOK = FALSE;
        $output = 'Database error: ' . $ex->getMessage() . ' in ' . $ex->getFile() . ":" . $ex->getLine();

    }

    if ($queryOK) {
        echo "Register deleted from Database<br>";
    } else { //Manejo errores
        echo "Please try again later<br>";
        echo "Debug information:<br>";
        echo $output . "<br>";
        echo $q . "<br>";
    }

?>
```

Pasar info por URL

Podemos pasar información de una web a otra, se debe emplear:

Origen

```
href='index.php?nombre='Pedro'
```

Destino

```
$_GET['nombre'];
```

Ejemplo completo SQL

```
<?php
require('./pdo_connect.php');

$q = "SELECT * FROM Nombres";

$result = $pdo->query($q);
$rows=$result->fetchAll();
foreach ($rows as $row) {
    echo $row['Nombre']."<br>";
}

if($_SERVER["REQUEST_METHOD"] == "POST")
if(isset($_POST['nombres'])) {
    $nombre = $_POST['nombres'];

    $queryOK=TRUE;
    try {
        $q = "INSERT INTO Nombres (nombre) VALUES (:nombre)";

        $stmt = $pdo->prepare($q);
        $stmt->bindValue(':nombre',$nombre);
        $stmt->execute();
    } catch(PDOException $ex) {
        $queryOK=FALSE;
        $output = 'Database error: ' . $ex->getMessage() . ' in ' . $ex->getFile() .
        ':' . $ex->getLine();
    }
    if ($queryOK) {
        echo "Register added to Database<br>";
    } else {
        echo "Please try again later<br>";
        echo "Debug information:<br>";
        echo $output."<br>";
        echo $q."<br>";
    }
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <br><br><br>
    <form action="" method="post">
        <input type="text" name="nombres">
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

```
~~~~~
```