



MEJORA CONTINUA EN LOS DESARROLLOS

El desarrollo web sufre constantes cambios y los sitios siempre presentan posibilidades de mejora. Aplica las siguientes medidas para asegurar la vigencia de las páginas, alargar su ciclo de vida; o bien, realizar mejoras a su funcionamiento.

Emplea la notación estándar en las etiquetas de **apertura y cierre del código PHP**, así aseguras el **funcionamiento de la aplicación** en cualquier servidor.

<?php y ?>

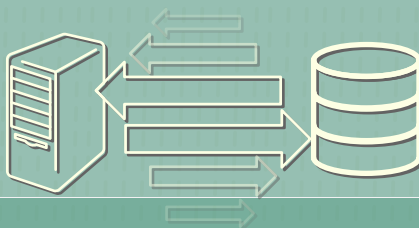
Utiliza la función **echo** para la **impresión de mensajes** en la página. Esta función es la más eficaz para realizar esta tarea y **disminuye el tiempo** de procesamiento del programa.

 **<? echo "Función para escribir"; ?>**

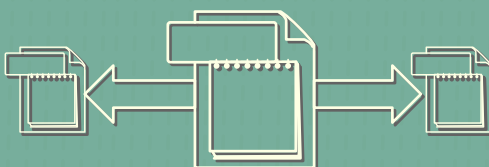
Evita escribir **código HTML** dentro del **bloque de PHP**, así separas el **funcionamiento operacional** del diseño del sitio.

```
<?
echo "<html>";
echo "<head>";
echo "</head>";
echo "<body>";
    echo "Un mensaje en PHP";
echo "</body>";
echo "</head>";
?>
```

Reduce el número de consultas a la base de datos para evitar **procesos redundantes** y codificar programas escalables.




Separa el código y **reutiliza** las funciones que ya tengas implementadas, de esta manera podrás **ahorrar tiempo** y esfuerzo al momento de desarrollar tus programas.





MEJORA CONTINUA EN LOS DESARROLLOS

Define todas las variables al principio del código, evita la declaración de variables globales y **no declares variables** que no se utilicen en la ejecución del programa.




```
<?
$variable 1 = 0;
$variable 2 = 3;
$variable 3 = "Suma";
$resultado;
.
.
.
$resultado = $variable1 + $variable2;
echo "La" + $variable3 + "es:" + $resultado + "";
```

Emplea el cotejamiento adecuado para **utilizar caracteres especiales** en la codificación del sitio, de lo contrario, la página **mostrará caracteres distintos** al momento de imprimir un acento o un símbolo desconocido.



{utf-8}

Verifica la información de los formularios mediante la **función isset**, de esta manera aseguras que los datos de ingreso no estén vacíos. También puedes utilizar la función **mysql_real_escape_string** para saber si los datos contienen caracteres que pueden provocar un **ataque al sitio**.



Estructura el código en bloques para que puedas **identificar** fácilmente los bloques de ejecución del programa.



```
<html>
  <head>
    ....
  </head>
  <body>
    ....
    <?php
      escho"....";
    ?>
    ....
  </body>
</html>
```