

EJEMPLOS Y EJERCICIOS PHP.CURSO

Profesor: Ing. Nestor Huaracha Velasquez

Ej



FUNDAMENTOS DE PHP CON HTML

Contenido

PRACTICA 1.....	7
PRACTICA 2.....	7
PRACTICA 3.....	7
PRACTICA 4.....	8
Ejercicio 1 - 01_info.php.....	8
Ejercicio 2 - 02_hola.php.....	9
Ejercicio 3 - 03_hola2.php.....	9
PRACTICA 5.....	10
Ejercicio 4 - 04_variable1.php.....	11
Ejercicio 5 - 05_variable2.php.....	12
Ejercicio 6 - 06_break.php.....	13
Ejercicio 7 - 07_break2.php.....	14
Ejercicio 8 - 08_weekdays1.php.....	15
Ejercicio 9 - 09_weekdays2.php.....	16
Ejercicio 10 - 10_Operaciones.php.....	17
Ejercicio 11 - 11_Comparacion.php.....	18
Ejercicio 12 - 12_Logicos.php.....	19
Ejercicio 13 - 13_Ejer01.php.....	20
Ejercicio 14 - 14_Redondeo.php.....	21
Ejercicio 15 - 15_if.php.....	22
Ejercicio 16 - IF..Else.....	23
PRACTICA 6.....	23
PRACTICA 7.....	23
PRACTICA 8.....	24
PRACTICA 9.....	24
Ejercicio 17 - 16_switch.php.....	24
Ejercicio 18 - 17_Tabla1.php.....	25
Ejercicio 19 - 18_tabla2.php.....	27
Ejercicio 20 - 19_tabla3.php.....	28
Ejercicio 21 - 20_while.php.....	29
20.html.....	30
Ejercicio 22 - 20_whileB.php.....	31
PRACTICA 10.....	32
Ejercicio 23 - 21_libreria.phtml.....	32
22_pag1.php.....	33
23_pag2.php.....	33
Ejercicio 24 - 24_analisis.php.....	34
Ejercicio 25 - 25_analisis.php.....	36
Ejercicio 26 - 26_analisis.php.....	37
Ejercicio 27 - 27_pass1.php.....	38

Newpage1.html	39
Newpage2.html	39
Sorry.html	39
Ejercicio 28 - 28_feedback.php	40
Ejercicio 29 - 29_counter.php	42
Ejercicio 30 - 30_guestbook.php	43
Ejercicio 31 - 31_encuesta.php	45
Ejercicio 32 - 32_readfeeds	46
PRACTICA 10 – Formulario (controles text y submit)	47
PRACTICA 11 – Formulario (control radio)	47
PRACTICA 12 - Formulario (control checkbox).	47
PRACTICA 13 – Formulario (control select)	47
PRACTICA 14 – Formulario (control textarea)	48
PRACTICA 15 – Vectores (tradicionales)	48
PRACTICA 16 – Creacion de un archivo de texto	48
PRACTICA 17 – Lectura de un archivo de texto	49
PRACTICA 18 – Vectores (asociativos)	49
PRACTICA 19 – Funciones	49
Ejercicio 33 - 33_seguridad	51
Ejercicio 34 - 34_album.php	52
Ejercicio 40 - 40_mysql01.php	54
Ejercicio 41 - 41_mysql02.php	55
Ejercicio 42 - 42_mysql03.php	56
Acceso.inc.php	57
Ejercicio 43 – Alta de registros	58
pagina2.php	58
PRACTICA 20 – Insert (alta de registros en una tabla)	59
Ejercicio 44 – Listado	59
PRACTICA 21 – Listado (seleccion de registros de una tabla)	60
Ejercicio 45 – Consulta	60
pagina2.php	60
PRACTICA 22 – Consulta (selección de registros de una tabla)	61
Ejercicio 46 – Delete	61
pagina2.php	61
PRACTICA 23 – Delete (Baja de un registro en una tabla)	62
Ejercicio 47 – Delete (todo)	62
PRACTICA 24 – Delete (Baja de todos los registros de una tabla)	63
Ejercicio 48 – Update	63
pagina2.php	63
pagina3.php	64
PRACTICA 25 – Update (Modificacion de un registro de una tabla)	64
Ejercicio 49 – Insert	65
pagina2.php	65
PRACTICA 26 – Insert (y consulta de otra tabla)	66

Ejercicio 50 – Listado 2 tablas	66
PRACTICA 27 – Listado (selección de registros de varias tablas – INNER JOIN)	67
Ejercicio 51 – Funcion count.....	67
PRACTICA 28 – Funcion count de SQL.....	67
Ejercicio 52 – Update 2 tablas	68
pagina2.php	68
pagina3.php	69
PRACTICA 29 – Update (modificacion de un registro trabajando con dos tablas)	69
Ejercicio 53 – Group by.....	70
PRACTICA 30 – Clausula Group by de SQL.....	70
Ejercicio 54 – Parametros en hipervinculo.....	71
pagina2.php	71
PRACTICA 31 – Parametros en un hipervinculo	72
Ejercicio 55 – Paginacion	72
PRACTICA 32 – Paginacion de registros.....	73
Ejercicio 56 – Subir un archivo	73
pagina2.php	74
PRACTICA 33 – Subir un archivo al servidor (upload).....	74
Ejercicio 57 – Cookies.....	74
pagina2.php	75
PRACTICA 34 – Creacion y lectura de una cookie.....	76
Ejercicio 58 – Borrado de cookie.....	76
pagina2.php	76
PRACTICA 35 – Borrado de una cookie.....	77
Ejercicio 59 – Cookie de sesion	77
pagina2.php	78
Ejercicio 60 – Variables de sesion	78
pagina2.php	79
pagina3.php	79
PRACTICA 36 – Variables de sesion (\$_SESSION)	80
Ejercicio 61 – Incluir un archivo externo	80
pagina2.php	80
PRACTICA 37 – Incluir un archivo externo (require_once).....	81
Ejercicio 62 – Redireccionar a otra pagina	81
pagina2.php	82
PRACTICA 38 – Redireccionar a otra pagina (header)	82
Ejercicio 63 – Creacion de imágenes dinamicas.....	82
PRACTICA 39 – Creacion de imágenes dinamicas desde PHP.....	83
Ejercicio 64 – Captcha en HTML.....	84
pagina2.php	84
pagain3.php	85
PRACTICA 40 – Agregar imágenes dinamicas en un archivo HTML.....	85
Ejercicio 65 – Fechas	86
pagina2.php	86

pagina3.php	87
PRACTICA 41 – Administracion de fechas y horas (funcion date)	87
Ejercicio 66 – Validacion de una fecha ingresada	88
66_checkdate1.php	88
66_checkdate2.php	88
PRACTICA 42 – Validacion de una fecha ingresada por teclado (checkdate)	89
Ejercicio 67 – Carga de una fecha en una tabla	89
PRACTICA 43 – Carga de una fecha en una tabla de MySQL	91
Ejercicio 68 – Funcion printf()	91
PRACTICA 44 – Formateo de datos en una pagina (printf)	93
Ejercicio 69 – Funcion sprintf()	93
PRACTICA 45 – Formateo de datos y salida a un string (sprintf)	94

PARTE I



PRACTICA 1

Para la lección 1.1 deberás leer las páginas de la 4 a la 10 del Cuaderno de Trabajo. A continuación responde lo que se te pide:

PREGUNTA	RESPUESTA
¿Quién inventó PHP?	Rasmus Lerdorf
¿Qué es PHP?	PHP es un lenguaje de programación que se usa principalmente para crear páginas web dinámicas. Permite que las páginas web muestren contenido diferente según la información que recibe del servidor, como bases de datos o archivos. Es popular por ser fácil de usar y se mezcla
¿Qué significa PHP?	Hypertext Preprocessor
¿Cuáles son las diferencias más notables entre JavaScript y PHP?	JavaScript es un lenguaje de programación del lado del cliente, mientras que PHP es un lenguaje de programación del lado del servidor. JavaScript se ejecuta en el navegador web del cliente, mientras que PHP se ejecuta en el servidor web. JavaScript se utiliza principalmente para la interactividad en el navegador, como validación de formularios, efectos visuales, etc., mientras que PHP se utiliza para la lógica del servidor, como acceder a bases de datos, procesar formularios, generar contenido dinámico, etc.
¿Qué tipo de licencia tiene PHP?	PHP utiliza la licencia PHP License, que es una licencia de código abierto compatible con la Licencia Pública General de GNU (GPL)
¿Dónde podemos encontrar ayuda para php?	puedes consultar la documentación oficial en php.net, participar en comunidades en línea como Stack Overflow, leer libros y tutoriales, y seguir blogs y sitios web especializados en PHP.
¿Qué necesitamos para trabajar con PHP?	necesitas un servidor web que admita PHP
¿Cómo accedamos a nuestro servidor local?	XAMPP, WAMP o MAMP
¿Dónde se encuentra la página de inicio de WAMP?	C:\wamp\www

PRACTICA 2

Investiga sobre las diferentes versiones de php existentes y sus características

PRACTICA 3

Realiza la práctica de Instalación WAMP en tu Pc (Busca en internet y descarga el WAMP. También deberás configurarlo)

PREGUNTA	RESPUESTA
¿De que sitio obtuviste el WAMP?	Sin embargo, WAMP (Windows, Apache, MySQL, PHP/Perl/Python)
¿Qué versión del WAMP obtuviste?	3.3.2
¿Qué servicios instalaste del WAMP?	phpMyAdmin

PRACTICA 4

Realiza la práctica de Configuración de Wamp.

Carga la página principal de wampserver con un navegador web que tenga;

Cree una Carpeta y asígnele un nombre dentro de la dirección localhost/www/xxxxxxx para almacenar todos los ejercicios de PHP.

Instala y/o inicia un editor de código ej. Notepad++, etc.

Ejercicio 1 - 01_info.php

NOTA: Primer programa en php, para mostrar como se escribe el código dentro de una página

```
<html>
<head>
  <title>PHP-Info</title>
</head>

<body>

<?php
phpinfo();
?>

</body>
</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué es lo que hace la función phpinfo()?	La función phpinfo() en PHP muestra información detallada sobre la configuración de PHP en el servidor donde se ejecuta. Al llamar a esta función en un script PHP y luego ejecutar ese script en el servidor, se genera una página web que enumera diversas configuraciones de PHP, incluyendo la versión de PHP, la configuración del servidor, las extensiones habilitadas, la configuración de la directiva php.ini y mucha más información útil sobre el entorno de PHP

Ejercicio 2 - 02_hola.php

NOTA: El famoso programa HOLA MUNDO en php

```
<html>

<head>
  <title>Hola Mundo</title>
</head>

<body>
<h1> El famoso script Hello World</h1>
<p>
<?php

echo "Hola Mundo!";

?>
</p>
</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Cómo comenzamos un bloque de php?	<pre><?php // Tu código PHP aquí echo "¡Hola, mundo!"; ?></pre>
¿Cómo imprimimos un mensaje en pantalla?	<pre><?php print "¡Hola, mundo!"; ?></pre>

Ejercicio 3 - 03_hola2.php

NOTA: Variacion del programa anterior, usando ademas etiquetas HTML para darle enfasis

```
<html>

<head>
  <title>Hola Mundo</title>
</head>

<body>
<h1> El famoso script Hello World</h1>
<p>
```

```
<?php  
  
echo "<b>Hola</b> Mundo!";  
  
?>  
</p>  
</body>  
  
</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Cuál es la diferencia con el ejercicio 2?	El ejercicio 2 no es una función sino una construcción del lenguaje que se utiliza para imprimir texto en pantalla. Aquí tienes una explicación detallada de la diferencia

PRACTICA 5

Confeccione un programa que muestre una serie de mensajes en la página empleando el comando ECHO. Tenga en cuenta que cuando utiliza el comando ECHO el mensaje se debe encerrar entre comillas dobles.

```
<!DOCTYPE html>  
<html lang="es">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Mensajes con Echo</title>  
</head>  
<body>  
  <h1>Mensajes con Echo</h1>  
  <?php  
  // Mensajes a mostrar  
  $mensaje1 = "Este es el primer mensaje.";   
  $mensaje2 = "Este es el segundo mensaje.";   
  $mensaje3 = "Y este es el tercer mensaje.";   
  
  // Mostrar mensajes usando echo  
  echo "<p>$mensaje1</p>";  
  echo "<p>$mensaje2</p>";  
  echo "<p>$mensaje3</p>";  
  ?>  
</body>  
</html>
```

Ejercicio 4 - 04_variable1.php

NOTA: Uso de variables en php y como se declaran

```
<html>

<head>
  <title></title>
</head>

<body>
<h1>Primer ejemplo de Variables </h1>
<p>
<?php
$Name = "Miguel";
echo "Hola <b>$Name</b>, encantado de conocerte";

?>
</p>
</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Cómo declaramos una variable?	\$nombre_de_la_variable;
¿Importan las mayúsculas y minúsculas en la declaración de las variables?	En PHP, el nombre de una variable distingue entre mayúsculas y minúsculas, lo que significa que las mayúsculas y minúsculas importan en la declaración de las variables. Esto significa que \$mensaje, \$Mensaje y \$MENSAJE se considerarían variables diferentes en PHP.

Ejercicio 5 - 05_variable2.php

NOTA: Uso de variables en variacion al programa anterior, usando concatenacion de variables

```
<html>

<head>
  <title></title>
</head>

<body>
<h1>Segundo ejemplo de Variables usando el operador concatenacion .</h1>
<p>
<?php
$name = "Miguel";
echo "Hola <b>" . $name . "</b>, encantado de conocerte";

?>
</p>
</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué es la concatenacion?	La concatenación es un término utilizado en programación para referirse a la operación de unir dos o más cadenas de caracteres en una sola cadena más larga. En PHP, puedes concatenar cadenas utilizando el operador de concatenación, que es un punto (.).
¿Cómo hacemos la concatenacion en php?	<pre>\$nombre = "Juan"; \$apellido = "Pérez"; \$nombre_completo = \$nombre . " " . \$apellido; echo \$nombre_completo; // Esto imprimirá: Juan Pérez</pre>

Ejercicio 6 - 06_break.php

NOTA: Ejemplo para ver como se formatea el código para el usuario

```
<html>

<head>
  <title></title>
</head>

<body>
<h1>Break en la ventana del navegador </h1>
<p>
<?php
$Name = "Miguel";
echo "Hola <b>$Name</b>, encantado de conocerte<br>";
echo "Gracias por venir!";
?>
</p>
</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué es lo que hace el script?	El script PHP dado toma dos variables, \$nombre y \$apellido, y luego las concatena para formar una cadena que representa el nombre completo. Luego, utiliza la función echo para imprimir esta cadena en la página web.

Ejercicio 7 - 07_break2.php

NOTA: Ejemplo de variación del script anterior, para ver el formato dentro de php en avance de línea. Notar las diferencias cuando se haga la ejecución vía VER CODIGO FUENTE del navegador.

```
<html>

<head>
  <title></title>
</head>

<body>
<h1>Break en la ventana del navegador con signo \ </h1>
<p>
<?php
$Name = "Miguel";
echo "Hola <b>$Name</b>, encantado de conocerte<br>\n";
echo "Gracias por venir!\n";
?>
</p>
</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Cuál es la diferencia con el ejercicio 6?	<p>La diferencia principal entre el ejercicio 6 y el ejercicio anterior es la forma en que se realiza la concatenación de las cadenas para formar el nombre completo.</p> <p>En el ejercicio 6, se utiliza el operador de concatenación (.) para unir las variables \$nombre y \$apellido, como se muestra a continuación:</p>

Ejercicio 8 - 08_weekdays1.php

NOTA: Aunque los arrays se ven mas adelante según el temario del silabo, en el cuaderno de trabajo vienen antes y no son dificiles de asimilar. Queda a discrecion ponerlos en orden o hacerlos despues del no. 21 que son los ciclos y antes de entrar a los formularios.

```
<html>

<head>
  <title>Introduccion a los arrays, metodo largo</title>
</head>

<body>
<h1> Introduccion a los arrays, metodo largo </h1>
<p> A continuacion escribiremos los arrays de acuerdo al metodo largo </p>
<p>
<?php
/*
Este es el metodo largo para la creacion de arrays, donde cada uno de los
arreglos esta en una linea diferente, junto con su valor indice
correspondiente
*/
$dia[0] = "domingo";
$dia[1] = "lunes";
$dia[2] = "martes";
$dia[3] = "miercoles";
$dia[4] = "jueves";
$dia[5] = "viernes";
$dia[6] = "sabado";
//mostrar el miercoles

echo $dia[3];

?>
</p>
</body>
</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué es un array?	Un array es una estructura de datos que puede contener múltiples valores, llamados elementos, cada uno identificado por un índice. En PHP, un array puede contener elementos de diferentes tipos de datos, como enteros, cadenas, booleanos u otros arrays.
¿Cómo declaramos un array?	\$numeros = array(1, 2, 3, 4, 5);

Ejercicio 9 - 09_weekdays2.php

NOTA: Es el mismo ejercicio de arreglos anterior, pero aquí se ven en una sola línea para simplificar el código. También, hay que hacer notar el uso de comentarios y como se ponen en php.

```
<html>

<head>
  <title>Introduccion a los arrays, metodo corto</title>
</head>

<body>
<h1> Introduccion a los arrays, metodo corto </h1>
<p> A continuacion escribiremos los arrays de acuerdo al metodo corto </p>
<p>
<?php
/*
Este es el metodo corto para la creacion de arrays, donde cada uno o todo el
array puede estar en una sola linea de codigo.
*/
$dia = array( "domingo", "lunes", "martes", "miercoles", "jueves", "viernes", "sabado");
//mostrar el miercoles

echo $dia[0];

?>
</p>
</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué diferencia hay con el script 7?	El script 7 utiliza una variable llamada \$nombre_completo para almacenar una cadena que representa el nombre completo de una persona, mientras que el script 6 (suponiendo que te refieres al script 6) utiliza un array para almacenar el nombre y el apellido por separado.
¿Cómo se ponen los comentarios?	// Este es un comentario de una línea símbolo //////////

Ejercicio 10 - 10_Operaciones.php

NOTA: Uno de los temas es ver como se realizan las operaciones en php, en este ejemplo se ven cadenas y numeros y como se hacen las operaciones aritmeticas basicas, asi como los incrementos y decrementos a las variables. Tambien vemos como se crea una variable en php.

```
<html>

<head>
  <title>Ejemplo de operaciones</title>
</head>

<body>
<h1>Ejemplo de operaciones aritmeticas en PHP</h1>
<?php

$a = 8;
$b = 3;
echo $a + $b, "<br>";
echo $a - $b, "<br>";
echo $a * $b, "<br>";
echo $a / $b, "<br>";

$a++;
echo $a,"<br>";
$b--;
echo $b,"<br>";

?>

</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Cómo se declara una variable numerica en php?	\$edad = 25; // Variable que representa la edad de una persona (entero)
¿Cómo se realiza una operación con variables numericas?	\$numero1 = 10; \$numero2 = 20; \$suma = \$numero1 + \$numero2; echo "La suma es: " . \$suma; // Imprime: La suma es: 30
¿Es posible mostrar en pantalla en una misma linea cadenas y numeros?	\$edad = 25; echo "La edad de la persona es: " . \$edad;
¿Cómo incrementamos una variable?	\$numero = 5; \$numero++; // Incrementa \$numero en 1 echo \$numero; // Imprime: 6

Ejercicio 11 - 11_Comparacion.php

NOTA: Sobre el mismo punto de operaciones, pero ahora de comparacion con variables numericas y observar el resultado en tiempo de ejecucion.

```
<html>

<head>
  <title>Ejemplo de operadores de Comparacion</title>
</head>

<body>
<h1>Ejemplo de operaciones comparacion en PHP</h1>
<?php

$a = 8;
$b = 3;
$c = 3;
echo $a == $b, "<br>";
echo $a != $b, "<br>";
echo $a < $b, "<br>";
echo $a > $b, "<br>";
echo $a >= $c, "<br>";
echo $a <= $c, "<br>";

?>

</body>

</html>
```

Anota el significado de las operaciones de comparación:

PREGUNTA	RESPUESTA
==	Igualdad
!=	Diferente
<	Menor que
>	Mayor que
>=	Mayor o igual que
<=	Menor o igual que

Ejercicio 12 - 12_Logicos.php

NOTA: Por último, las operaciones lógicas se manejan según este ejemplo en php.

```

/* 12_logicos.php
Programa de ejemplo de operadores logicos
Por: Lic. Sergio Hugo Sanchez O.
Para: Universidad Matamoros
17, Mayo, 2009 */

<html>

<head>
  <title>Ejemplo de operadores Logicos</title>
</head>

<body>
<h1>Ejemplo de operaciones logicas en PHP</h1>
<?php

$a = 8;
$b = 3;
$c = 3;
echo ($a == $b) && ($c > $b), "<br>";
echo ($a == $b) || ($b == $c), "<br>";
echo !($b <= $c), "<br>";

?>

</body>

</html>

```

Anota el significado de las operaciones de comparacion:

PREGUNTA	RESPUESTA
&&	AND lógico
	OR lógico

Ejercicio 13 - 13_Ejer01.php

NOTA: Este es un ejercicio para los alumnos el cual deben realizarlo por ellos mismos, comparar sus resultados con este ejercicio resuelto para ver si entendieron las bases de php, comandos, operaciones, variables, uso de html.

RESULTADO:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Calculadora PHP</title>
</head>
<body>
  <h2>Calculadora PHP</h2>
  <form method="post" action="">
    <label for="numero1">Número 1:</label>
    <input type="number" id="numero1" name="numero1" required>
    <br><br>
    <label for="numero2">Número 2:</label>
    <input type="number" id="numero2" name="numero2" required>
    <br><br>
    <input type="submit" value="Sumar">
  </form>

  <?php
  if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Recibe los valores del formulario
    $numero1 = $_POST["numero1"];
    $numero2 = $_POST["numero2"];

    // Realiza la suma
    $suma = $numero1 + $numero2;

    // Muestra el resultado
    echo "<p>La suma de $numero1 y $numero2 es: $suma</p>";
  }
  ?>
</body>
</html>
```

Ejercicio 14 - 14_Redondeo.php

NOTA: Ejemplo para cálculos en php usando variables, formateando el código del script y usando algunos comandos nuevos de php (similares a c++)

```
<!Ejemplo de redondeo>
<html>

<head>
  <title>Calculos </title>
</head>

<body>
<h1>Calculos, redondeo y formato. </h1>
<?php
  /* Primero declaramos las variables */
  $precioneto = 101.98;
  $iva = 0.196;
  $resultado = $precioneto * $iva;
  echo "El precio es de ";
  echo $precioneto;
  echo " y el IVA el ";
  echo $iva;
  echo "% <br>";
  echo "Resultado: " ;
  echo round($resultado,2);
  echo " con ROUND() <br>";
  echo $resultado;
  echo " normal \n";
  echo "<br><br>";
  $resultado2 = sprintf("%01.2f", $resultado);
  echo "Usando la funcion SPRINTF se ve asi: ";
  echo $resultado2
?>

</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué funciones de PHP usamos en el script?	<code>\$_SERVER["REQUEST_METHOD"]</code>

Ejercicio 15 - 15_if.php

NOTA: Ejemplo del uso de la sentencia IF en la comparativa de variables

```
<html>

<head>
  <title>Condicion IF</title>
</head>

<body>
<h1>Condicional IF</h1>
<?php
  $a = 8;
  $b = 3;
  echo "Valor de a: ",$a," y el valor de b:",$b,"<br>";
  if ($a<$b)
  {
    echo "a es menor que b";
  }
  else
  {
    echo " a no es menor que b";
  }

?>

</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué hace la sentencia IF...ELSE?	se utiliza para ejecutar bloques de código diferentes dependiendo de si una condición es verdadera o falsa. Aquí está la sintaxis general de la sentencia if...else

Ejercicio 16 – IF..Else

NOTA: Uso de la sentencia If..else y una funcion de fecha.

```
<html>
<head></head>
<body>

<?php
$dia=date("d");
if ($dia<=10)
{
    echo "sitio activo";
}
else
{
    echo "sitio fuera de servicio";
}
?>

</body>
</html>
```

PRACTICA 6

Sabiendo que la funcion RAND nos retorna un valor aleatorio entre un rango de dos enteros:

```
$num = rand(1,100);
```

En la variable \$num se almacena un valor entero que la computadora genera en forma aleatoria entre 1 y 100. Hacer un programa que lo muestre por pantalla al valor generado. Mostrar ademas si es menor o igual a 50 o si es mayor. Para imprimir el contenido de una variable tambien utilizamos el comando ECHO

PRACTICA 7

Definir una variable de cada tipo: integer, double, string y boolean. Luego imprimirlas en la pagina, una por linea.

PRACTICA 8

Definir tres variables enteras. Luego definir un string que incorpore dichas variables y las sustituya en tiempo de ejecución. Recordar que una variable se sustituye cuando el string esta encerrado por comillas dobles:

```
$precio = 90;
```

```
Echo "La computadora tiene un precio de $precio";
```

PRACTICA 9

Generar un valor aleatorio entre 1 y 5. Luego imprimir en castellano el numero (Ej. Si se genera el 3 luego mostrar en la pagina el string "tres"). Para ver si una variable es igual a cierto valor debemos plantear una condicion similar a:

```
If ($valor==3)
```

```
{  
    //algoritmo  
}
```

Ejercicio 17 - 16_switch.php

NOTA: Uso de la sentencia SWITCH según el valor de una condicion de una variable, uso de comandos para salirse del codigo.

```
<html>  
  
<head>  
  <title>Condicional Switch</title>  
</head>  
  
<body>  
<h1>Condicional Switch. Ejemplo</h1>  
<?php  
  /*Declaramos una variable con un valor de muestra */  
  $posicion = "arriba";  
  echo "La variable posicion es ",$posicion;  
  echo "<br>";  
  
  switch($posicion){  
    case "arriba":      // Primer condicion si es arriba  
      echo "La variable contiene el valor de arriba";
```

```
        break;
    case "abajo":          //Segunda condicion del supuesto
        echo "La variable contiene el valor de abajo";
        break;
    default:              //Condicion por default o si no es ninguna
        echo "La variable contiene otro valor distinto arriba y abajo";
}

?>

</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Cuál es la diferencia de la instrucción SWITCH con la instrucción IF?	<p>Uso de casos específicos:</p> <p>La instrucción if evalúa una expresión booleana y ejecuta un bloque de código si la expresión es verdadera.</p> <p>La instrucción switch evalúa una expresión y ejecuta bloques de código diferentes según el valor de la expresión.</p>

Ejercicio 18 - 17_Tabla1.php

NOTA: Ejemplo del uso de ciclos for, funciones, comandos html, variables y tablas para combinar php y html.

```
<html>

<head>
  <title>Tabla condicional</title>
</head>

<body>
<h1>Tabla condicional 1</h1>
<?php
  /* Crearemos una tabla de valores de seno y coseno de 0 a 2
  en incrementos de 0.01. Los valores negativos que resulten los queremos
  mostrar en rojo, y los valores positivos en azul */
  /* En primer lugar vamos a crear una funcion con las condicionales
```

Aqui estamos usando la sentencion de FUNCTION al que le pasamos un valor dependiendo de este valor, ejecutamos una condicion IF..ELSE... y asignamos un color al tipo de letra para generar la tabla

```
*/
function muestra($valor) {
  if ($valor < 0.5)
    $color = "red";
  else
    $color = "blue";
  echo "<td><font color='$color'>$valor</font></td>\n";
}

?>
<table border="1">
<?
  for ($x=0; $x<=2; $x+=0.01){
    echo "<tr>";
    muestra($x);
    muestra(sin($x));
    muestra(cos($x));
    echo "</tr>";
  }
?>
</body>
</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
Explica que es lo que hace el script	<p>Este script es un ejemplo simple de cómo imprimir el nombre de un día de la semana en base a un número del 1 al 7 que representa el día. El script tiene dos partes principales: una que utiliza la instrucción if y otra que utiliza la instrucción switch.</p> <p>Parte con if:</p> <p>En esta parte del script, se define una variable \$dia que representa el número del día de la semana.</p> <p>Luego, se utiliza la estructura de control if...elseif...else para evaluar el valor de \$dia.</p> <p>Cada bloque if verifica si el valor de \$dia coincide con un número específico del 1 al 7, que corresponde a un día de la semana.</p> <p>Si el valor de \$dia coincide con uno de los números del 1 al 7, se imprime el nombre correspondiente del día de la semana utilizando la función echo.</p> <p>Si el valor de \$dia no coincide con ningún número del 1 al 7, se imprime un mensaje indicando que el número de día es inválido.</p>

Ejercicio 19 - 18_tabla2.php

NOTA: Variación del ejercicio anterior, dando color a cada reglón par e impar, mediante nuevas instrucciones.

```
<html>
<head>
  <title>Tabla condicional 2</title>
</head>
<body>
<h1>Tabla condicional 2</h1>
<?php
  /* Crearemos una tabla de valores de seno y coseno de 0 a 2
  en incrementos de 0.01. Los valores negativos que resulten los queremos
  mostrar en rojo, y los valores positivos en azul */

  /* Variacion. Un color diferente cada fila que se imprima */
  function muestra($valor) {
    if ($renglon % 2)
      $fondo = "#eeeeee";
    else
      $fondo = "#dddddd";

    if ($valor < 0.5)
      $color = "red";
    else
      $color = "blue";
    echo "<td bgcolor='$fondo'><font color='$color'>$valor</font></td>\n";
  }

  ?>
<table border="1">
<?
  $nrenglon = 0;
  for ($x=0; $x<=2; $x+=0.01){
    echo "<tr>";
    muestra($x);
    muestra(sin($x));
    muestra(cos($x));
    echo "</tr>";
  }
  ?>
</body>
</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Cuál es la diferencia con el script anterior?	El script anterior utiliza la estructura de control <code>if...elseif...else</code> para realizar esta tarea, mientras que el script proporcionado utiliza la estructura de control <code>switch</code> .

Ejercicio 20 - 19_tabla3.php

NOTA: Variación del ejemplo anterior usando variables globales. Investigar en internet o texto y plasmar en su cuaderno como se realiza este ejemplo

```
<html>
<head>
  <title>Tabla condicional 3</title>
</head>
<body>
<h1>Tabla condicional 3</h1>
<?php
  /* Crearemos una tabla de valores de seno y coseno de 0 a 2
  en incrementos de 0.01. Los valores negativos que resulten los queremos
  mostrar en rojo, y los valores positivos en azul */

  /* Variacion. Un color diferente cada fila que se imprima
  Para que funcione bien, necesitamos que la variable renglon se tome
  "desde afuera" y no de manera local dentro de la funcion*/
  function muestra($valor) {
    global $nrenglon;
    if ($nrenglon % 2)
      $fondo = "#ffff00";
    else
      $fondo = "#ffffff";

    if ($valor < 0.5)
      $color = "red";
    else
      $color = "blue";
    echo "<td bgcolor='$fondo'><font color='$color'>$valor</font></td>\n";
  }

?>
```

```
<table border="1">
<?
    $nrenglon = 0;
    for ($x=0; $x<=2; $x+=0.05){
        echo "<tr>";
        muestra($x);
        muestra(sin($x));
        muestra(cos($x));
        echo "</tr>";
    }
?>
</body>
</html>
```

Ejercicio 21 - 20_while.php

NOTA: Uso de ciclo while para hacer una repetición tantas veces quiera el usuario, para ello trabajamos con un formulario externo vinculado a php para hacerlo. Mostrar la conexión y variables externas, así como el uso del while.

```
<html>

<head>
    <title>Bucle While</title>
</head>

<body>
<h1>Bucle While</h1>
<?php
    /* Mostraremos el uso de la sentencia While y comenzamos a usar entrada
    del teclado mediante un formulario simple */
    if ( isset( $_POST['number'] ) ) {
        $number = $_POST['number'];
        $counter = 1;
        while ( $counter <= $number ) {
            echo "Los bucles son faciles!<br>\n";
            $counter++;
        }
        echo "Se acabo.\n";
    }
}
```

```
?>
</p>
</body>

</html>
```

20.html

```
<html>
<head>
  <title></title>
</head>
<body>
<form action="20_while.php" method="post">
¿Cuántas veces?
<input type="text" name="number">
<input type="submit" value="Enviar">
</form>
</body>
</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Por qué tenemos dos archivos en este ejercicio, uno php y el otro html?	El uso de dos archivos, uno PHP y otro HTML, permite separar la lógica de programación (PHP) de la presentación visual (HTML). Esto hace que el código sea más organizado, fácil de mantener y permite la colaboración entre diferentes equipos de desarrollo. El archivo PHP maneja la lógica del programa, mientras que el archivo HTML define la estructura y el diseño visual de la página web. Esta separación facilita la reutilización de código, el mantenimiento y las actualizaciones.
¿Qué es un formulario?	Un formulario web es una herramienta que permite a los usuarios ingresar datos y enviarlos al servidor para su procesamiento. Consiste en campos donde los usuarios pueden escribir texto, seleccionar opciones, marcar casillas, etc. Estos formularios son esenciales para interactuar con los usuarios en sitios web y recopilar información de ellos, como en registros, búsquedas, compras en línea, entre otros.
Explica el ciclo While	El ciclo `while` ejecuta un bloque de código repetidamente mientras una condición sea verdadera.
Que hace \$Counter++	incrementa el valor de la variable \$Counter en uno.

Ejercicio 22 - 20_whileB.php

NOTA: Variacion sobre el script anterior, pero usando un solo archivo para poner juntos el formulario y el codigo php. Opcional si los alumnos realizan solos este ejercicio o lo explica el maestro según su disposicion y entendimiento.

```
<html>

<head>
  <title></title>
</head>

<body>
<h1>Bucle While 2</h1>
<form action="20_whileB.php" method="post">
¿Cuantas veces?
<input type="text" name="number">
<input type="submit" value="Enviar">
</form>
<p>
<?php
  /* Mostraremos el uso de la sentencia While y comenzamos a usar entrada
  del teclado mediante un formulario simple */
  if ( isset( $_POST['number'] ) ) {
    $number = $_POST['number'];
    $counter = 1;
    while ($counter <= $number) {
      echo "<b>$counter</b>.- Los bucles son faciles!<br>\n";
      $counter++;
    }
    echo "Se acabo.\n";
  }
?>
</p>
</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Cuál es la diferencia con el script anterior?	El script anterior utiliza la estructura de control <code>if</code> , mientras que este utiliza la estructura <code>switch</code> para realizar la misma tarea.

PRACTICA 10

Mostrar la tabla de multiplicar del 2. Emplear el for, luego el while y por ultimo el do/while. La estructura FOR permite incrementar una variable de 2 en 2:

```
For ($f=2; $f<=20; $f=$f+2)
```

Ejercicio 23 - 21_libreria.phtml

NOTA: Uso de funciones en php y como se vinculan con otros archivos php. Notar la extensión de la función en el primer archivo y como se incrustan estas librerías de funciones en otros scripts de php para ser usados.

```
<html>

<head>
  <title>Uso de Librerías y Funciones</title>
</head>

<body>
  <?php
    function CabeceraPagina()
    {
  ?>
    <FONT SIZE="+1">Esta cabecera estará en todas sus páginas.</FONT><BR>
    <hr>
  <?
    }

    function PiePagina()
    {
  ?>
    <hr>
    <FONT SIZE="-1">Este es el pie de página.</FONT><BR>
    Autor: Joaquin Gracia
  <?
    }
  ?>

</body>
</html>
```

22_pag1.php

```
<html>

<head>
  <title>Ejemplo Pagina 1 Libreria</title>
</head>

<body>
<?php include("21_libreria.phtml") ?>
<?php CabeceraPagina(); ?>

Página 1
<BR><BR><BR><BR><BR>

Contenido blalbl blalb alb<BR><BR>
más cosas...<BR><BR>

fin<BR><BR>

<?php PiePagina(); ?>

</body>

</html>
```

23_pag2.php

```
<html>

<head>
  <title>Ejemplo Pagina 2 Librerias</title>
</head>

<body>
<?php include("21_libreria.phtml") ?>
<?php CabeceraPagina(); ?>

Esta es otra página<BR><BR>
completamente distinta<BR><BR>
pero comparte el pie y la cabecera con la otra.<BR><BR>

<?php PiePagina(); ?>
```

```
</body>
</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué es lo que hace este script?	21_libreria.phtml y 23_pag2.php, dos librerías
¿Cómo creamos una función Personalizada?	<pre>function nombreDeLaFuncion(\$parametro1, \$parametro2) { // Cuerpo de la función // Haz lo que necesites con los parámetros }</pre>
¿Cómo insertamos una función?	Para insertar una función en PHP, define la función usando <code>function nombreDeLaFuncion() { // código de la función }</code> y luego llama a la función usando <code>nombreDeLaFuncion()</code> .
¿Cómo creamos un archivo de funciones o librería de funciones?	<code>include('mis_funciones.php');</code>

Ejercicio 24 - 24_analisis.php

NOTA: Primer ejercicio con formularios, hacer notar como se pasa la variable del formulario al script php.

```
<html>

<head>
  <title>EJERCICIO 3_1: ANALISIS DE FORMULARIO</title>
</head>

<body>
<h1>EJERCICIO 3_1: ANALISIS DE FORMULARIO</h1><br>
<br>
<h1>analisis de formularios (analisis.php)</h1>
<?php

    echo "Hola <b>{$_POST['firstname']}</b>, encantado de saludarte.";
?>

</body>
```

</html>

24_analisis.htm

<html>

<head>

<title>EJERCICIO 3_1: ANALISIS DE FORMULARIO</title>

</head>

<body>

<h1>EJERCICIO 3_1: ANALISIS DE FORMULARIO</h1>

<h1>Rellena los campos (form.html)</h1>

<form action="24_analisis.php" method="post"> Tu nombre: <input type="text" name="firstname">

<input type="submit">

</form>

</body>

</html>

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué hace la variable \$_POST?	'\$_POST' en PHP es un array asociativo que contiene datos enviados desde un formulario HTML utilizando el método POST. Las claves del array son los nombres de los campos del formulario y los valores son los datos ingresados por el usuario.
¿Por qué usamos el metodo POST en los formularios?	Usamos el método POST en los formularios para enviar datos desde el cliente al servidor de una manera segura y eficiente, especialmente cuando se envían grandes cantidades de datos o datos sensibles.

Ejercicio 25 - 25_analisis.php

NOTA: Variación del formulario anterior con más variables por pasar.

```
<html>
<head>
  <title>EJERCICIO 3_1: ANALISIS DE FORMULARIO</title>
</head>
<body>
<h1>EJERCICIO 3_1: ANALISIS DE FORMULARIO</h1><br>
<br>
<h1>analisis de formularios (analisis.php)</h1>
<?php
  if($_POST['gender'] == 0) {
    echo "Hola Sr. ";
  } else {
    echo "Hola Sra. ";
  }

  echo "<b>{$_POST['lastname']}</b>, encantado de saludarte.";
?>
</body>
</html>
```

25_analisis.html

```
<html>
<head>
  <title>EJERCICIO 3_1: ANALISIS DE FORMULARIO Version 2</title>
</head>
<body>
<h1>EJERCICIO 3_1: ANALISIS DE FORMULARIO Version 2</h1><br>
<br>
<h1>Rellena los campos (form.html)</h1>
<form action="25_analisis.php" method="post">
  <input type="radio" name="gender" value="0"> Sr.
  <input type="radio" name="gender" value="1"> Sra.<br>
  Tu apellido:<br>
  <input type="text" name="lastname">
  <input type="submit">
</form>
</body>
</html>
```

Ejercicio 26 - 26_analisis.php

NOTA: Variación del formulario análisis poniendo el código html del formulario y el código del script php en un mismo archivo.

```
<html>

<head>
  <title>EJERCICIO 3_2</title>
</head>

<body>
<h1>EJERCICIO 3_1: ANALISIS DE FORMULARIO Version 3 Todo en una pagina</h1><br>
<br>
<h1>Rellena los campos (form.html)</h1>
<form action="25_analisis.php" method="post">
  <input type="radio" name="gender" value="0"> Sr.
  <input type="radio" name="gender" value="1"> Sra.<br>
  Tu apellido:<br>
  <input type="text" name="lastname">
  <input type="submit" name="submitButton" value="Envialo!">
</form>
<?php
  if (isset($_POST['gender']) && isset($_POST['lastname']) && $_POST['lastname'] != "") {
    if($_POST['gender'] == 0) {
      echo "Hola Sr. ";
    } else {
      echo "Hola Sra. ";
    }
    echo "<b>".$_POST['lastname'].</b>, encantado de saludarte.\n";
  } else {
    if (isset($_POST['submitButton'])) {
      echo "Por favor rellena todos los campos" ;
    }
  }
?>

</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué hace la funcion isset()?	La función `isset()` en PHP se utiliza para comprobar si una variable está definida y no es nula. Retorna `true` si la variable está definida y tiene un valor distinto de `null`, y `false` si la variable no está definida o su valor es `null`.
¿Cuáles son los operadores relacionales que estamos usando?	Los operadores relacionales que estamos usando son `==` y `!=`. Estos operadores se utilizan para comparar si dos valores son iguales o diferentes, respectivamente.

Ejercicio 27 - 27_pass1.php

NOTA: Este programa 27 es para poner password a archivos html. El primer archivo lleva código php con el password puesto en el encabezado antes de iniciar el código html, para redirigir a los archivos correctos o incorrectos. Luego viene el formulario que pide el password, de acuerdo al password dado, se dirige a newpage1 o newpage2 según sea el caso. Si no se sabe el password correcto se va a un tercer archivo llamado sorry.

```
<?php

if (isset($_POST['pw'])) {
    $pw = $_POST['pw'];
    if ($pw == "magic") {
        header ("Location: newpage1.html");
    } elseif ($pw == "abracadabra" ){
        header ("Location: newpage2.html");
    } else {
        header ("Location: sorry.html");
    }
}

?>

<html>

<head>
    <title>Ejemplo de password y header</title>
</head>

<body>
<h1> Ejemplo de password y funcion header </h1>
<form action="<?php echo $_SERVER['PHP_SELF']; ?>" method="post">
<input type="text" name="pw">
```



```
<input type="submit" value="Envialo">
</form>
```

```
</body>
```

```
</html>
```

Newpage1.html

```
<html>
<head>
  <title>newpage1</title>
</head>
```

```
<body>
Esta es la pagina protegida con MAGIC
newpage1.html
</body>
</html>
```

Newpage2.html

```
<html>

<head>
  <title>newpage2</title>
</head>
```

```
<body>
Esta es la pagina protegida con ABRACADABRA
newpage2
</body>

</html>
```

Sorry.html

```
<html>

<head>
  <title>Sorry</title>
</head>

<body>
Lo sentimos, ese no es el password correcto. Intentalo otra vez
sorry.html
```

```
</body>
```

```
</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Por qué se pone el código php antes del código html?	El código PHP se coloca antes del código HTML en un archivo PHP para que PHP pueda procesar y generar el contenido dinámico antes de que se envíe al navegador del usuario. Esto permite una separación clara entre la lógica de programación (PHP) y la presentación visual (HTML).
¿Qué hace \$_SERVER?	`\$_SERVER` en PHP es una variable superglobal que contiene información sobre el entorno del servidor web y la solicitud HTTP. Se utiliza para acceder a una variedad de información, como la dirección IP del cliente, el nombre del servidor, la URL solicitada, entre otros detalles relacionados con la solicitud y el servidor.
¿Es recomendable poner las claves de esta forma en php?	No, no es recomendable poner claves de esta manera en PHP ya que se pueden exponer datos sensibles y comprometer la seguridad de la aplicación.

Ejercicio 28 - 28_feedback.php

NOTA: Pequeño script que muestra un formulario que se enviara por correo electrónico al web master todo en un solo archivo. Indicar a los alumnos que es lo que hace cada línea para que la desglosen e indiquen lo aprendido.

```
<html>
<head>
<title> un pequeño mailer para recopilar la opinión </title>
<meta http-equiv="content-type" content="text/html; charset=ISO-8859-1" >
</head>
<body>
<h1>Feedback-Mailer</h1>
<p>¡Envíame un e-mail!</p>
<form action="<?php echo $_SERVER['PHP_SELF']; ?>" method="post">
Tu dirección de e-mail: <br>
<input type="text" name="Mail" ><br>
Tu comentario: <br>
<textarea name="message" cols="50" rows="5">
```

```

</textarea><br>
<input type="submit" value="Enviar">
</form>
<?php
$receiverMail = "tudireccion@tudominio.es"; // escribe aqui tu dirección
if (isset($_POST['Mail']) && $_POST['Mail'] != "") {
    if (mail ($receivermail, "¡Tienes correo nuevo!", $_POST['message'], "From:
    $_POST[Mail]")) {
        echo "<p>Gracias por enviarme tu opinión.</p>\n";
    } else {
        echo "<p>Lo siento, ha ocurrido un error.</p>\n";
    }
}
?>
</body>
</html>

```

Investigar más variación del script en el Cuaderno con FORMMAILER UNIVERSAL.

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Es importante poner la línea 4 del script en nuestro código?	Sí, es importante poner la línea 4 del script en nuestro código porque incluir el archivo de funciones nos permite utilizar las funciones definidas en él en nuestro script.
¿Funciona bien este programa?	Si funciona

Ejercicio 29 - 29_counter.php

NOTA: Un contador de visitas para cualquier página web. Es necesario crear un archivo de texto en blanco llamado counter.txt en la misma ubicación donde se ejecuta el script.

```
<html>
<head>
  <title>Contador Sencillo</title>
</head>

<body>
<h1>Contador Sencillo</h1>
<p>Cantidad de visitas:
<b>
<?php
  // Para el contador necesitamos un archivo de texto externo donde almacenamos
  // las visitas
  // Aqui se verán las funciones para trabajar con archivos, en este caso
  // abrir, leer, grabar y cerrar. El contador simplemente se va sumando.

  $fp = fopen("counter.txt", "r+");
  $counter = fgets($fp, 7);
  echo $counter;
  $counter ++;
  rewind($fp);
  fputs($fp, $counter);
  fclose($fp);

?>
</b></p>
</body>

</html>
```

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué hace la función fopen()?	La función `fopen()` en PHP se utiliza para abrir un archivo en modo de lectura o escritura, y devuelve un puntero de archivo que se puede utilizar para realizar operaciones de lectura o escritura en el archivo.
¿Qué hace fgets()?	`fgets()` en PHP se utiliza para leer una línea de un archivo abierto previamente con `fopen()`. Devuelve la línea como una cadena de caracteres o `false` si se alcanza el final del archivo.
¿Qué hace rewind()?	`rewind()` en PHP se utiliza para volver al principio de un archivo

	abierto previamente con <code>`fopen()'</code> . Esto permite volver a leer el archivo desde el principio después de haberlo leído o escrito.
¿Qué hace <code>fputs()</code> ?	<code>`fputs()'</code> en PHP se utiliza para escribir una cadena en un archivo abierto previamente con <code>`fopen()'</code> . Es equivalente a la función <code>`fwrite()'</code> .
¿Qué hace <code>fclose()</code> ?	<code>`fclose()'</code> en PHP se utiliza para cerrar un archivo que ha sido abierto previamente con <code>`fopen()'</code> . Esto libera los recursos asociados al archivo y lo hace inaccesible para futuras operaciones de lectura o escritura.
¿Por qué necesitamos <code>counter.txt</code> ?	Necesitamos el archivo <code>`counter.txt'</code> para almacenar el número de visitas a nuestra página web. Cada vez que alguien visita la página, PHP lee el valor almacenado en <code>`counter.txt'</code> , lo incrementa en uno y luego guarda el nuevo valor de vuelta en el archivo. Esto nos permite llevar un registro del número total de visitas a nuestra página.

Ejercicio 30 - 30_guestbook.php

NOTA: Libro de visitas basado en php. Es necesario crear un archivo de texto llamado guestbook.txt para almacenar los comentarios de los visitantes.

El código es bastante largo y contiene varios comentarios para facilitar la lectura del mismo.

```
<html>
<head>
<title>Un libro de visitas muy sencillo</title>
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
</head>
<body>
<h1>Libro de visitas</h1>
<form action="<?php echo $_SERVER['PHP_SELF']; ?>" method="post"
Tu comentario:<br>
<textarea cols="55" rows="4" name="comment"></textarea><br>
Tu nombre:<br>
<input type="text" name="name"><br>
Tu e-mail:<br>
<input type="text" name="email"><br>
<input type="submit" value="publica:">
</form>
<h3>Mostrar todos los comentarios</h3>
<?php
//Guarda el nombre del archivo en la variable
$file = "guestbook.txt";
//¿variable comment definida? ¿Nombre e e-mail no estan vacios?
//if isset($_POST['comment']) && $_POST['name'] != "" && $_POST['email'] != "" {
if (isset($_POST['comment']) && isset($_POST['name']) != "" && $_POST['email'] != "") {

    $comment = $_POST['comment'];
    $name = $_POST['name'];
    $email = $_POST['email'];

    //El archivo se abre para escritura-lectura
    $fp = fopen($file, "r+");
    //Leer todos los datos y almacenar en $old
```

```

    $old = fread ($fp, filesize($file));
    //Se crea el vinculo de e-mail
    $email = "<a href=\mailto:$email\">$email</a>";
    //Se incluye la fecha y se le da formato
    $dateOfEntry = date ("y-n-j");
    //Ocultar caracteres html, eliminar slashes, mantener saltos de linea
    $comment = htmlspecialchars($comment);
//    $comment = stripslashes(n12br($comment));
    $comment = stripslashes($comment);
    //"Montar la entrada (entry) del libro de visitas
    $entry="<p><b>$name</b> ($email) wrote on <i>$dateOfentry</i>;<br>$comment</p>\n";
    //El cursor invisible salta al principio
    rewind($fp);
    //Escribir en la nueva entrada antes de las antiguas en el archivo:
    fputs($fp, "$entry \n $old");
    //cerrar el archivo
    fclose($fp);
}
//Mostrar el archivo completo
readfile($file);
?>
</body>
</html>

```

Ver la variación con protección contra RELOAD, además de la prevención esta la averiguación del IP del cliente y las funciones DATE

Responde a lo siguiente:

PREGUNTA	RESPUESTA
¿Qué hace htmlspecialchars()?	Escapa caracteres.
¿Qué hace stripslashes()?	Elimina barras.
¿Qué hace readfile()?	Lee archivo.

Ejercicio 31 - 31_encuesta.php

NOTA: Hacer un script para un POLL o ENCUESTA que según la decisión de los usuarios seleccionen de un tema. Se necesita un archivo de texto llamado RESULTS.TXT para almacenar lo que responden los usuarios.

Código largo para hacer.

```
<?php
setcookie("check", 1);
if (isset($_POST['submit'])) {
    setcookie("vote", 1);
}
?>
<html>
<head>
<title>Encuesta de opinion</title>
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
</head>
<body>
<h1>Encuesta</h1>
<h3>¿Que opinas de este curso de php?</h3>
<form action="<?php echo $_SERVER['PHP_SELF']; ?>" method="post">
<input type="radio" name="reply" value="0">
Excelente, he aprendido mucho.<br>
<input type="radio" name="reply" value="1">
Mas o menos, es muy complicado.<br>
<input type="radio" name="reply" value="2">
¡Bah! para que quiero aprender php
<br> <br>
<?php
if (empty($_POST['submit']) && empty($_COOKIE['voted'])) {
//Mostrar el botón submit solo si el formulario todavia
// no se ha enviado y el usuario no ha votado.
?>
<input name="submit" type="submit" value="vota!">
<?php
} else {
    echo "<p>Gracias por tu voto.</p>\n";
    //¿Formulario enviado? ¿cookies activas? ¿pero todavia no se ha votado?
    if (isset($_POST['reply']) && isset($_COOKIE['check']) && empty($_COOKIE['voted'])) {
        //Guardar nombre de archivo en la variable
        $file="results.txt";
        $fp=fopen($file, "r+");
```



```
$vote=fread($fp, filesize($file));  
//Descomponer la string del archivo en array con coma como separador  
$arr_vote = explode(",", $vote); //explode convierte la string en array  
//¿que valor se ha seleccionado en el formulario?  
//¡El recuento aumenta en 1!  
$reply = $_POST['reply'];  
$arr_vote[$reply]++;  
//volver a montar la string  
$vote = implode(",", $arr_vote); // implode vincula elementos de la array a string  
rewind($fp);  
//Escribir nueva string en el archivo  
fputs($fp, $vote);  
fclose($fp);  
}  
?  
</form>  
<p>  
[ <a href="results.php" target="_blank">ver resultados de la encuesta</a>]  
</p>  
</body>  
</html>
```

Ejercicio 32 - 32_readfeeds

Investigar y Realizar este ejercicio en su cuaderno de trabajo.

PRACTICA 10 – Formulario (controles text y submit)

Confeccionar un formulario que solicite la carga de un nombre de persona y su edad, luego mostrar en otra pagina si es mayor de edad (si la edad es mayor o igual a 18).

PRACTICA 11 – Formulario (control radio)

Solicitar que se ingrese por teclado el nombre de una persona y disponer tres controles de tipo radio que nos permitan seleccionar si la persona:

- 1) no tiene estudios
- 2) estudios primarios
- 3) estudios secundarios

En la pagina que procesa el formulario mostrar el nombre de la persona y un mensaje indicando el tipo de estudios que posee.

PRACTICA 12 - Formulario (control checkbox)

Confeccionar un formulario que solicite la carga del nombre de una persona y que permita seleccionar una serie de deportes que practica (futbol, basket, tennis, voley).

Mostrar en la pagina que procesa el formulario la cantidad de deportes que practica

PRACTICA 13 – Formulario (control select)

Confeccionar un formulario que solicite el ingreso del nombre de una persona y un combo de selección (en este ultimo permitir la selección de los ingresos anuales de la persona: 1-1000, 1001-3000, >3000).

En la pagina que procesa el formulario mostrar un mensaje si debe pagar impuestos a las ganancias (si supera 3000).

PRACTICA 14 – Formulario (control textarea)

Confeccionar una pagina que muestre un contrato dentro de un textarea, disponer puntos suspensivos donde el operador debe ingresar un texto. La pagina que procesa el formulario solo debe mostrar el contrato con las modificaciones que hizo el operador.

Ej. De un contrato puede ser:

En la ciudad de [.....], se acuerda entre la Empresa [.....] representada por el Sr. [.....] en su carácter de Apoderado, con domicilio en la calle [.....] y el Sr. [.....], futuro empleado con domicilio en [.....], celebrar el presente contrato a Plazo Fijo, de acuerdo a la normativa vigente de los articulos 90, 92, 93, 94, 95 y concordantes de la Ley de Contrato de Trabajo No. 20744.

PRACTICA 15 – Vectores (tradicionales)

Definir un vector con los nombres de los dias de la semana. Luego imprimir el primero y el ultimo elemento del vector.

PRACTICA 16 – Creacion de un archivo de texto

Confeccionar un programita en PHP que permita hacer el pedido de pizzas via internet. El formulario debe ser:

Nombre:
[.....]
Direccion:
[.....] Jamon
y queso [x]
Cantidad [....]

Napolitana
[x]
Cantidad
[....]

Mozzarella
[x]
Cantidad
[....]

[Confirmar]

Para el ingreso del nombre, direccion y cantidad de pizzas de cada tipo disponer objetos de la clase “text”. Disponer tres objetos de tipo “check” para seleccionar los tipos de pizzas. Por ultimo disponer un boton para el envio de datos: “submit”.

Grabar en un archivo de texto cada pedido, separados por una línea de puntos (obligatoriamente dar el nombre de texto como "datos.txt", esto es para que no se llene el disco duro de archivos.

PRACTICA 17 – Lectura de un archivo de texto

Confeccionar un programa que muestre el archivo de pedido de pizzas via internet anteriormente creado. (Obligatoriamente dar el nombre de archivo de texto como "datos.txt").

PRACTICA 18 – Vectores (asociativos)

Crear un vector asociativo que almacena las claves de acceso de 5 usuarios de un sistema. Acceder a cada componente por su nombre. Imprimir un componente por cada vector.

PRACTICA 19 – Funciones

Confeccionar un formulario que solicite la carga del nombre de usuario y su clave en dos oportunidades. En la página que se procesan los datos del formulario implementar una función que imprima un mensaje si las dos claves ingresadas son distintas.

PARTE II

Php y MySql

Ejercicio 33 - 33_seguridad

NOTA: Este es el primer ejercicio del segundo cuaderno o manual del Curso: PHP y MySQL (PC Cuadernos No.30). A partir de aquí se manejarán los ejercicios de este cuaderno. Aquí se muestra una falla en la seguridad de PHP cuando tenemos el `register_globals = ON` en la configuración de `php.ini`.

Para ver que es fácil entrar a la página protegida con password, basta teclear en el navegador:

http://localhost/33_seguridad.php?login=1

Y entraremos en el área “protegida” por el programa. Así cualquiera puede entrar.

```
<html>
<head>
  <title>El gran agujero de seguridad</title>
</head>
<body>
<h2>Agujero de seguridad en register_globals = On</h2>
<form action="<?php echo "33_seguridad.php"; ?>" method="post">
Contraseña: <input type="password" name="pass">
<input type="submit" value="Enviar">
</form>
<?php
  if ($pass == "abc_xyz_123") {
    $login = true;
  }
  if ($login) {
    echo "<p>Aqui empieza el arma secreta.</p>";
  }
?>
</body>
</html>
```

Ejercicio 34 - 34_album.php

NOTA: Este ejemplo muestra como crear un album de fotos, asi como algunas funciones mas avanzadas de php que pueden ser consultadas en el manual de php oficial. El programa subira a un directorio especificado una imagen seleccionada por el usuario siempre que sea GIF o JPG y no pase de 200kb, a este directorio y luego mostrara todas las imágenes que se hayan subido.

Contiene varias sentencias de condicion, funciones de php para manejo de archivos y directorios y un formulario, asi como variables internas y externas.

Mas adelante viene una extension de este programa como prueba final.

```
<html>
<head>
  <title>Mi album de fotografias</title>
</head>

<body>
<h1> Mi album de fotos en linea </h1>
<h3> Cargar archivo </h3>

<form action='<?php echo $_SERVER['PHP_SELF'] ?>' method="post" enctype="multipart/form-data">
<input type="file" name="archivo">
<input type="submit" name="submit" value="Cargar archivo">
</form>

<?php
$ruta="img/"; // Indicar ruta
if (isset($_FILES['archivo']) && $_FILES['archivo']['size'] > 0) {
  $tamanyomax = 200000; // Indicar tamaño en bytes
  $nombretemp = $_FILES['archivo']['tmp_name'];
  $nombrearchivo = $_FILES['archivo']['name'];
  $tamanyoarchivo = $_FILES['archivo']['size'];
  $tipoarchivo = GetImageSize( $nombretemp );
  if ($tipoarchivo[2] == 1 || $tipoarchivo[2] == 2) { //Gif o Jpg?
    if (tamanyoarchivo <= $tamanyomax) { // Archivo demasiado grande?
      if (move_uploaded_file($nombretemp, $ruta.$nombrearchivo)) {
        echo "<p>El archivo se ha cargado <b>con exito</b>.";
        Tamaño de archivo: <b>$tamanyoarchivo</b> bytes,
        Nombre de imagen: <b>$nombrearchivo</b><br></p>";
      } else {
        echo "<p>No se ha podido cargar el archivo.</p>";
      }
    }
  }
}
```



```
    }
    } else {
        echo "<p>El archivo tiene mas de <b>$tamanyomax bytes</b> y es demasiado grande.</p>";
    }
    } else {
        echo "<p>No es un archivo GIF o JPG valido.</p>";
    }
    echo "<form action='{$_SERVER['PHP_SELF']}' method='post'>
        <input type='submit' value='OK'></form>";
}
$filehandle = opendir($ruta); // Abrir archivos
while ($file=readdir($filehandle)) {
    if ($file != "." && $file != "..") {
        $tamanyo = GetImageSize($ruta.$file);
        echo "<p><img src='$ruta$file' $tamanyo[3]><br></p>\n";
    }
}
closedir($filehandle); // Fin lectura de archivos
?>

</body>
</html>
```

Ejercicio 40 - 40_mysql01.php

NOTA: Este es el primer ejercicio de MySQL y php donde se mostraran los datos almacenados en una tabla de MySQL. Esta base de datos y tabla debio haberse creado primero con phpmyadmin e introducir unos datos de ejemplo en cada campo.

Tambien puede usarse el administrador de XAMPP o en su caso HeidiSQL para accesar a la creacion de la base de datos, tablas, campos e introducir la informacion de ejemplo.

La base de datos se llama AGENDA, y la tabla se llama DIRECCIONES. Que contiene solamente 2 campos: Nombre y Apellido.

Aquí veremos las funciones basicas de conexión a la base de datos con php `mysql_connect`, `mysql_select_db`, `mysql_query`, `mysql_fetch_assoc`, `mysql_close`

```
<html>
<head>
  <title>MySQL 01 - Consulta a BD (Agenda)</title>
</head>
<body>
<h1>Mostrar Nombres de la Agenda. BD </h1>
<?php
  $dp = mysql_connect("localhost", "root", "");
  mysql_select_db("agenda", $dp);
  $sql = "SELECT * FROM direcciones" ;
  $resultado = mysql_query($sql);
  while ($row = mysql_fetch_assoc($resultado)) {
    echo "$row[Nombre] $row[Apellido]<br>\n";
  }
  mysql_close($dp);
?>
</body>
</html>
```

Ejercicio 41 - 41_mysql02.php

NOTA: En esta version del primer programa, mostramos la informacion de la agenda pero usando tablas o acomodado en las tablas para darle una mejor presentacion. Aquí cambiamos el ciclo while por un for

```
<html>
<head>
  <title>MySQL 02 - Consulta BD con tabla (Agenda)</title>
</head>
<body>
<h1>MySQL 02 - Consulta BD con tabla (Agenda)</h1>
<?php
  $dp = mysql_connect("localhost", "root", "");
  mysql_select_db("agenda", $dp);
  $sql = "SELECT * FROM direcciones" ;
  $resultado = mysql_query($sql);
  $campos = mysql_num_fields($resultado);
  $filas = mysql_num_rows($resultado);
  echo "<p>Cantidad de filas: $filas</p>\n";
  echo "<table border='1' cellspacing='0'>\n"; //Empezar tabla
  echo "<tr>"; //Crear fila
  for ($i = 0; $i < $campos; $i++) {
    $nombrecampo = mysql_field_name($resultado, $i);
    echo "<th>$nombrecampo</th>";
  }
  echo "</tr>\n"; //Cerrar fila

  while ($row = mysql_fetch_assoc($resultado)) {
    echo "<tr>"; //Crear fila
    foreach ($row as $key => $value) {
      echo "<td>$value&nbsp;</td>";
    }
    echo "</tr>\n"; //Cerrar fila
  }
  echo "</table>\n"; //Cerrar tabla
  mysql_close($dp);
?>
</body>
</html>
```

Ejercicio 42 - 42_mysql03.php

NOTA: Aquí tenemos un programa para introducir datos (Altas) a la base de datos mediante un formulario y las instrucciones básica de conexión MySQL y otras funciones de php ya vistas. Además se incluye otro archivo (con INCLUDE) que es quien realiza la conexión a la BD.

```
<html>

<head>
  <title></title>
</head>

<body>
<h3>Introducir direcciones</h3>
<?php
include("acceso.inc.php");
if (isset($_POST['submit'])) {
  if (empty($_POST['Nombre'])) {
    echo "<p>Introduzca el <b>nombre</b>.</p>";
  }
  else if (strlen($_POST['Apellido']) < 3) {
    echo "<p>El apellido debe tener como minimo <b>3</b> caracteres.</p>";
  }
  else {
    $sql = "INSERT INTO direcciones (Tratamiento, Nombre, Apellido, Calle, CP, Localidad, Tel, Movil,
Mail, Website, Categoria, Notas ) VALUES ('$_POST[Tratamiento]',
'$_POST[Nombre]', '$_POST[Apellido]', '$_POST[Movil]', '$_POST[Mail]', '$_POST[Website]',
'$_POST[Categoria]', '$_POST[Notas]')";
  }
  $resultado = mysql_query($sql);
  if ($resultado)
  {
    echo "<p> Datos agregados con exito.</p>";
  } else {
    echo "<p> Datos <b>no</b> agregados.</p>";
  }
  echo "[ <a href='javascript:history.back()'>Volver</a> ] - [ <a href='$_SERVER[PHP_SELF]'> Introducir
nueva fila</a>]";
}
else {
  $sql2 = "SELECT * FROM categorias";
  $resultado2 = mysql_query($sql2);
  $campocat = "";
  while ($row = mysql_fetch_assoc($resultado2)) {
```

```

        $campocat .= "<option value='$row[Categoria]>\n";
    }
    //echo <<<formulario. porque el manual dice que va aqui?
}
//FORMULARIO;
mysql_close($dp);
?>
<form action= "{$_SERVER['PHP_SELF']}" method="post" >
    <table>
        <tr><td>Tratamiento:</td><td><select name="Tratamiento">
            <option>Sr.</option>
            <option>Sra.</option>
        </select></td></tr>
        <tr><td>Nombre:</td><td><input type="text" name="Nombre"></td></tr>
        <tr><td>Apellido:</td><td><input type="text" name="Apellido"></td></tr>
        <tr><td>Calle:</td><td><input type="text" name="Calle"></td></tr>
        <tr><td>CP:</td><td><input type="text" name="CP"></td></tr>
        <tr><td>Localidad:</td><td><input type="text" name="Localidad"></td></tr>
        <tr><td>Tel:</td><td><input type="text" name="Tel"></td></tr>
        <tr><td>Movil:</td><td><input type="text" name="Movil"></td></tr>
        <tr><td>E-mail:</td><td><input type="text" name="Mail"></td></tr>
        <tr><td>Website:</td><td><input type="text" name="Website"></td></tr>
        <tr><td>Categoria:</td><td><select
name="Categoria">{$campocat}</select></td></tr>
        <tr><td>Notas:</td><td><textarea cols="60" rows="4"
name="Notas"></textarea></td></tr>
        <tr><td><input type="submit" value="Introducir datos"
name="submit"></td></tr>
    </table>
</form>
</body>

</html>

```

Acceso.inc.php

```

<?php
$dp = @mysql_connect("localhost", "root", "") or die("<p>No se ha podido establecer la conexion con
MySQL.</p>");
@mysql_select_db("agenda", $dp) or die("<p>No se ha podido establecer la conexion con la base de
datos.</p>");
?>

```

Ejercicio 43 – Alta de registros

43_alumnos.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<h1>Alta de Alumnos</h1>
<form action="pagina2.php" method="post">
Ingrese nombre:
<input type="text" name="nombre"><br>
Ingrese mail:
<input type="text" name="mail"><br>
Seleccione el curso:
<select name="codigocurso">
<option value="1">PHP</option>
<option value="2">ASP</option>
<option value="3">JSP</option>
</select>
<br>
<input type="submit" value="Registrar">
</form>
</body>
</html>

```

pagina2.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80")
    or die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la seleccion de la base de datos");
mysql_query("insert into alumnos(nombre,mail,codigocurso) values
    ('$_REQUEST[nombre]','$_REQUEST[mail]','$_REQUEST[codigocurso]'",
    $conexion) or die("Problemas en el select".mysql_error());
mysql_close($conexion);
echo "El alumno fue dado de alta.";
?>
</body>
</html>

```

PRACTICA 20 – Insert (alta de registros en una tabla)

Ya existe una tabla llamada: CURSOS en la base de datos PHPFACIL. Efectuar el alta de la tabla. La tabla de cursos tiene la siguiente estructura:

```
CREATE TABLE 'cursos' (  
'codigo' int(11) NOT NULL auto_increment,  
'nombrecur' varchar(40) NOT NULL,  
PRIMARY KEY ('codigo') )
```

Ejercicio 44 – Listado

```
<html>  
<head>  
<title>Problema</title>  
</head>  
<body>  
<?php  
$conexion=mysql_connect("localhost","root","z80")  
    or die("Problemas en la conexion");  
mysql_select_db("phpfacil",$conexion)  
    or die("Problemas en la selección de la base de datos");  
$registros=mysql_query("select codigo,nombre, mail, codigocurso  
                        from alumnos",$conexion) or  
    die("Problemas en el select:".mysql_error());  
while ($reg=mysql_fetch_array($registros))  
{  
    echo "Codigo:". $reg['codigo']. "<br>";  
    echo "Nombre:". $reg['nombre']. "<br>";  
    echo "Mail:". $reg['mail']. "<br>";  
    echo "Curso:";  
    switch ($reg['codigocurso']) {  
        case 1:echo "PHP";  
            break;  
        case 2:echo "ASP";  
            break;  
        case 3:echo "JSP";  
            break;  
    }  
    echo "<br>";  
    echo "<hr>";  
}  
mysql_close($conexion);  
?>  
</body>  
</html>
```

PRACTICA 21 – Listado (seleccion de registros de una tabla)

Confeccionar un programa que recupere los datos de la tabla CURSOS de la base de datos PHPFACIL

Ejercicio 45 – Consulta

NOTA: El proceso de consulta de datos de una tabla es similar al del listado, la diferencia es que se muestra solo aquel que cumple la condicion por la que buscamos.

45_alumnos03.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese el mail del alumno a consultar:
<input type="text" name="mail">
<br>
<input type="submit" value="buscar">
</form>
</body>
</html>
```

pagina2.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select codigo,nombre, codigocurso
                        from alumnos where mail='".$_REQUEST[mail]'", $conexion) or
    die("Problemas en el select:".mysql_error());
if ($reg=mysql_fetch_array($registros))
{
    echo "Nombre:". $reg['nombre']. "<br>";
    echo "Curso:";
    switch ($reg['codigocurso']) {
        case 1:echo "PHP";
            break;
        case 2:echo "ASP";
            break;
        case 3:echo "JSP";
            break;
    }
}
```



```

}
else
{
    echo "No existe un alumno con ese mail.";
}
mysql_close($conexion);
?>
</body>
</html>

```

PRACTICA 22 – Consulta (selección de registros de una tabla)

Confeccionar un programa que permita ingresar el nombre de un alumno en un formulario, luego mostrar los datos del mismo (tener en cuenta que puede haber mas de un alumno con el mismo nombre)

Ejercicio 46 – Delete

NOTA: El objetivo de este punto es el borrado de un registro de una tabla. Para ello, implementaremos un algoritmo que solicite ingresar el mail de un alumno y posteriormente efecture su borrado.

La primera pagina es identica a la consulta, ya que debemos implementar un formulario que solicite la carga del mail del alumno. Por otro lado tenemos el archivo “pagina2.php” que se encarga de buscar el mail ingresado en el formulario y en caso que exista se procede a borrarlo.

46_alumnos04.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese el mail del alumno a borrar:
<input type="text" name="mail">
<br>
<input type="submit" value="buscar">
</form>
</body>
</html>

```

pagina2.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<?php

```

```

$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select codigo from alumnos
                        where mail='$_REQUEST[mail]'", $conexion) or
    die("Problemas en el select:".mysql_error());
if ($reg=mysql_fetch_array($registros))
{
    mysql_query("delete from alumnos where mail='$_REQUEST[mail]'", $conexion) or
        die("Problemas en el select:".mysql_error());
    echo "Se efectuó el borrado del alumno con dicho mail.";
}
else
{
    echo "No existe un alumno con ese mail.";
}
mysql_close($conexion);
?>
</body>
</html>

```

PRACTICA 23 – Delete (Baja de un registro en una tabla)

Confeccionar un programa que permita ingresar el nombre de un curso por teclado y posteriormente efectue el borrado de dicho registro en la tabla CURSOS. Mostrar un mensaje si no existe el curso.

Ejercicio 47 – Delete (todo)

NOTA: Para borrar todos los registros de una tabla debemos llamar al comando delete de SQL sin disponer la clausula WHERE.

47_alumnos05.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
mysql_query("delete from alumnos", $conexion) or
    die("Problemas en el select:".mysql_error());
echo "Se efectuó el borrado de todos los alumnos.";
mysql_close($conexion);
?>

```

```
</body>
</html>
```

PRACTICA 24 – Delete (Baja de todos los registros de una tabla)

Efectuar el borrado de todos los registros de la tabla CURSOS

Ejercicio 48 – Update

NOTA: De las actividades con tablas, esta es la mas larga. Vamos a resolverlo implementando tres paginas. La primera un formulario de consulta del mail de un alumno, la segunda otro formulario que nos permita cargar su mail modificado y la ultima registrara el cambio en la tabla.

48_alumnos06.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese el mail del alumno:
<input type="text" name="mail"><br>
<input type="submit" value="buscar">
</form>
</body>
</html>
```

pagina2.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select * from alumnos
                        where mail='$_REQUEST[mail]'", $conexion) or
    die("Problemas en el select:".mysql_error());
if ($reg=mysql_fetch_array($registros))
{
```

```

?>
<form action="pagina3.php" method="post">
  Ingrese nuevo mail:
  <input type="text" name="mailnuevo" value="<?php echo $reg['mail'] ?>">
  <br>
  <input type="hidden" name="mailviejo" value="<?php
  echo $reg['mail'] ?>">
  <input type="submit" value="Modificar">
</form>
<?php
}
else
  echo "No existe alumno con dicho mail";
?>
</body>
</html>

```

pagina3.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
  die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
  die("Problemas en la selección de la base de datos");
$registros=mysql_query("update alumnos
                        set mail='$_REQUEST[mailnuevo]'
                        where mail='$_REQUEST[mailviejo]'", $conexion) or
  die("Problemas en el select:".mysql_error());
  echo "El mail fue modificado con exito";
?>
</body>
</html>

```

PRACTICA 25 – Update (Modificación de un registro de una tabla)

Efectuar la modificación del nombre del curso de la tabla CURSOS. Para la búsqueda ingresar el código de curso.

Ejercicio 49 – Insert

NOTA: Ahora vamos a ver como resolver el problema del alta de un alumno seleccionando el curso de la tabla “Cursos”. Es decir, el formulario de carga de datos no es HTML puro ya que debemos cargar el control “select” con los datos de la tabla CURSOS.

49_alumnos07.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese nombre:
<input type="text" name="nombre"><br>
Ingrese mail:
<input type="text" name="mail"><br>
Seleccione el curso:
<select name="codigocurso">
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select codigo,nombrecur from cursos",$conexion) or
    die("Problemas en el select:".mysql_error());
while ($reg=mysql_fetch_array($registros))
{
    echo "<option value=\"".$reg[codigo]\">".$reg[nombrecur]</option>";
}
?>
</select>
<br>
<input type="submit" value="Registrar">
</form>
</body>
</html>
```

pagina2.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la seleccion de la base de datos");
mysql_query("insert into alumnos(nombre,mail,codigocurso) values
    ('$_REQUEST[nombre]', '$_REQUEST[mail]',
    $_REQUEST[codigocurso])", $conexion) or
    die("Problemas en el select".mysql_error());
```

```
mysql_close($conexion);
echo "El alumno fue dado de alta.";
?>
</body>
</html>
```

PRACTICA 26 – Insert (y consulta de otra tabla)

Confeccionar el alta de la tabla alumnos empleando controles de tipo “radio” para la selección del curso.

Ejercicio 50 – Listado 2 tablas

NOTA: Ahora veremos como imprimir todos los alumnos inscritos a los cursos junto al nombre del curso donde esta inscrito. Los datos se encuentran en las tablas “alumnos” y “cursos”.

50_alumnos08.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select alu.codigo as codigo,nombre,mail,codigocurso,
                        nombrecur
                        from alumnos as alu
                        inner join cursos as cur on cur.codigo=alu.codigocurso",
                        $conexion) or
    die("Problemas en el select:".mysql_error());
while ($reg=mysql_fetch_array($registros))
{
    echo "Codigo:".$reg['codigo']."<br>";
    echo "Nombre:".$reg['nombre']."<br>";
    echo "Mail:".$reg['mail']."<br>";
    echo "Curso:".$reg['nombrecur']."<br>";
    echo "<hr>";
}
mysql_close($conexion);
?>
</body>
</html>
```

PRACTICA 27 – Listado (selección de registros de varias tablas – INNER JOIN)

Confeccionar un programa que permita ingresar el código de un alumno y nos muestre su nombre, mail y nombre del curso en el cual está inscrito. Hacer un formulario donde se ingrese el código de alumno y otra página donde se muestren los datos respectivos. Mostrar un mensaje si no existe el código de alumno ingresado.

Ejercicio 51 – Función count

NOTA: Cuando necesitamos saber la cantidad de registros que cumplen una condición podemos utilizar la función COUNT(), por ejemplo, si deseamos conocer la cantidad de alumnos que hay en la tabla “alumnos”.

51_alumnos09.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexión");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select count(*) as cantidad
                        from alumnos",$conexion) or
    die("Problemas en el select:".mysql_error());
$reg=mysql_fetch_array($registros);
echo "La cantidad de alumnos inscriptos son :".$reg['cantidad'];
?>
</body>
</html>
```

PRACTICA 28 – Función count de SQL

Confeccionar un programa que muestre por pantalla los nombres de todos los cursos y al final la cantidad total de cursos.

Ejercicio 52 – Update 2 tablas

NOTA: Ahora complicaremos un poco la modificación de un registro consultando dos tablas.

Supongamos que un alumno desea cambiarse de curso, es decir, se inscribió en uno y quiere borrarse de ese e inscribirse en otro diferente. Debemos mostrar en un “select” el curso actual en el que está inscrito y los otros cursos disponibles en la tabla CURSOS. Para resolver este problema tenemos que plantear tres páginas, una donde ingresemos el mail del alumno, la segunda donde se pueda cambiar el curso y por última una que permita modificar la tabla ALUMNOS con el nuevo curso seleccionado.

52_alumnos10.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese el mail del alumno:
<input type="text" name="mail"><br>
<input type="submit" value="buscar">
</form>
</body>
</html>
```

pagina2.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexión");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select * from alumnos
                        where mail='$_REQUEST[mail]'", $conexion) or
    die("Problemas en el select:".mysql_error());
if ($regalu=mysql_fetch_array($registros))
{
?>
<form action="pagina3.php" method="post">
<input type="hidden" name="mailviejo" value="<?php
    echo $regalu['mail'] ?>">
<select name="codigocurso">
<?php
    $registros=mysql_query("select * from cursos",$conexion) or
        die("Problemas en el select:".mysql_error());
    while ($reg=mysql_fetch_array($registros))
    {
        if ($regalu['codigocurso']==$reg['codigo'])
            echo "<option value=\"".$reg['codigo']." selected>$reg[nombrecur]</option>";
        else
            echo "<option value=\"".$reg['codigo'].">$reg[nombrecur]</option>";
```



```

    }
    ?>
</select>
<br>
<input type="submit" value="Modificar">
</form>
<?php
}
else
    echo "No existe alumno con dicho mail";
?>
</body>
</html>

```

pagina3.php

```

<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("update alumnos
                        set codigocurso=$_REQUEST[codigocurso]
                        where mail='$_REQUEST[mailviejo]'", $conexion) or
    die("Problemas en el select:".mysql_error());
echo "El curso fue modificado con éxito";
?>
</body>
</html>

```

PRACTICA 29 – Update (modificación de un registro trabajando con dos tablas)

Confeccionar la modificación del mail, nombre y curso de la tabla ALUMNOS. Ingresar por teclado el código del alumno para su búsqueda.

Ejercicio 53 – Group by

NOTA: Entre las muchas posibilidades que nos brinda SQL, una es agrupar registros y obtener información resumida de tablas. En nuestro problema, un listado interesante sería mostrar “la cantidad de alumnos inscritos por cursos”. Para resolver de una manera sencilla esta situación, SQL nos permite agrupar los registros de la tabla “alumnos” por la columna “codigocurso” y contar la cantidad de registros que hay por cada código de curso igual.

53_alumnos11.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexión");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select count(alu.codigo) as cantidad,
                        nombrecur from alumnos as alu
                        inner join cursos as cur on cur.codigo=alu.codigocurso
                        group by alu.codigocurso", $conexion) or
    die("Problemas en el select:".mysql_error());
while ($reg=mysql_fetch_array($registros))
{
    echo "Nombre del curso:".$reg['nombrecur']."<br>";
    echo "Cantidad de inscritos:".$reg['cantidad']."<br>";
    echo "<hr>";
}
mysql_close($conexion);
?>
</body>
</html>
```

PRACTICA 30 – Clausula Group by de SQL

Confeccionar un programa que muestre el nombre del curso, la cantidad de inscritos y todos los inscritos a dicho curso. Repetir esto con todos los cursos. Es decir, en la página debe aparecer algo parecido a:

Nombre del curso: PHP

Cantidad de inscripciones: 3

Nombres: Martinez Luis – Rodriguez Pablo – Gonzalez Ana

Nombre del curso: JSP

Cantidad de inscripciones: 2

Nombres: Hernandez Hector – Roca Marta

Ejercicio 54 – Parametros en hipervinculo

NOTA: Hasta ahora hemos visto como enviar datos de una pagina a otra mediante formularios. Otra forma muy utilizada que complementa la anterior es como parametro en un hipervinculo.

Confeccionaremos una pagina que muestres tres hipervinculos, cada uno tiene por objetivo mostrar en una pagina las tablas de multiplicar del 2, del 3 o la del 5. La primer pagina es un archivo HTML puro, ya que solo disponemos las marcas de hipervinculos.

54_alumnos12.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<A href="pagina2.php?tabla=2">Tabla del 2</A> <br>
<A href="pagina2.php?tabla=3">Tabla del 3</A> <br>
<A href="pagina2.php?tabla=5">Tabla del 5</A>
</body>
</html>
```

pagina2.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
echo "Listado de la tabla del $_REQUEST[tbl] <br>";
for($f=1;$f<=10;$f++)
{
$val=$f*$_REQUEST['tbl'];
echo $val."-";
}
?>
</body>
</html>
```

PRACTICA 31 – Parametros en un hipervinculo

Confeccionar un programa que muestre una pagina con todos los nombres de la tabla CURSOS. Los nombres deben aparecer como hipervinculos a otra pagina que debera mostrar todos los alumnos inscritos a dicho curso. Como parametro en el hipervinculo, pasar el codigo del curso.

Ejercicio 55 – Paginacion

NOTA: En situaciones en las cuales una consulta retorna muchos datos, en vez de enviarlos todos al navegador, se puede enviar un conjunto limitado de registros. Luego, mediante hipervinculos, ver el resto de los datos. Por ejemplo, cuando hacemos busquedas con el Servidor Google, generalmente no nos retorna todas las direcciones donde se encuentran los resultados buscados, nos retorna paginas con 10 enlaces por pagina (pensemos el tiempo de transferencia si nos retornara 1,000,000 de enlaces).

55_alumnos13.php

```
<?php
if (isset($_REQUEST['pos']))
    $inicio=$_REQUEST['pos'];
else
    $inicio=0;
?>
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select alu.codigo as
                        codigo,nombre,mail,codigocurso,
                        nombrecur from alumnos as alu
                        inner join cursos as cur on cur.codigo=alu.codigocurso
                        limit $inicio,2", $conexion) or
    die("Problemas en el select:".mysql_error());
$impresos=0;
while ($reg=mysql_fetch_array($registros))
{
    $impresos++;
    echo "Codigo:".$reg['codigo']."<br>";
    echo "Nombre:".$reg['nombre']."<br>";
    echo "Mail:".$reg['mail']."<br>";
    echo "Curso:".$reg['nombrecur']."<br>";
    echo "<hr>";
}
mysql_close($conexion);
if ($inicio==0)
```

```

    echo "anteriores ";
else
{
    $anterior=$inicio-2;
    echo "<a href=\"paginal.php?pos=$anterior\">Anteriores </a>";
}
if ($impresos==2)
{
    $proximo=$inicio+2;
    echo "<a href=\"paginal.php?pos=$proximo\">Siguientes</a>";
}
else
    echo "siguientes";
?>
</body>
</html>

```

Hay muchas cosas importantes en este ejemplo, lo primero que vemos es el bloque que rescata a partir de qué registro ir mostrando:

```

if (isset($_REQUEST['pos']))
    $inicio=$_REQUEST['pos'];
else
    $inicio=0;
?>

```

PRACTICA 32 – Paginacion de registros

Confeccionar un programa que muestre los registros de la tabla CURSOS con paginas de 3 registros

Ejercicio 56 – Subir un archivo

NOTA: Una actividad comun en un sitio es poder almacenar un archivo en el servidor, mas comunmente conocido como upload. Se necesita en muchas ocasiones este algoritmo, por ejemplo para subir fotos, documentos, programas, etc. Se requieren dos paginas, una de ellas, un formulario donde seleccionamos el archivo a enviar y otra pagina donde se graba el archivo en el servidor.

56_alumnos14.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post" enctype="multipart/form-data">
Seleccione el archivo:
<input type="file" name="foto"><br>
<input type="submit" value="Enviar">
</form>

```

```
</body>
</html>
```

pagina2.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
copy($_FILES['foto']['tmp_name'],$_FILES['foto']['name']);
echo "La foto se registro en el servidor.<br>";
$nom=$_FILES['foto']['name'];
echo "<img src=\"\$nom\">";
?>
</body>
</html>
```

PRACTICA 33 – Subir un archivo al servidor (upload)

Confeccionar un programa que permita hacer multiples upload con una pagina (por ejemplo, que permita seleccionar hasta 3 archivos)

Ejercicio 57 – Cookies

NOTA: El protocolo HTTP es desconectado. Esto significa que cada vez que solicitamos una página a un servidor representa una conexión distinta.

Una cookie es una pequeña cantidad de datos almacenada por el navegador del usuario cuando solicita una página a un servidor. El que envía que se genere la cookie es el servidor.

Una cookie consta de un nombre, un valor, una fecha de expiración y un servidor. Una cookie está limitada a 4KB.

Luego que una cookie es creada sólo el sitio que la creó puede leerla. Luego de creada una cookie, cada vez que el navegador del usuario visita el sitio, se envía dicha cookie. Otra cosa importante que hay que tener en cuenta es que el usuario del browser puede configurar el mismo para no permitir la creación de cookies, lo que significa que el uso de cookies debe hacerse con moderación y cuando la situación lo requiera. De todos modos, el 95% de los navegadores están configurados para permitir la creación de cookies.

Para la creación de una cookie desde PHP debemos llamar a la función setcookie.

Los parámetros de esta función son:

setcookie(<nombre de la cookie>, <valor de la cookie>, <fecha de expiración>, <carpeta del servidor>)

Con un problema sencillo entenderemos el uso de esta función. Supongamos que queremos que los usuarios que entran a nuestro sitio puedan configurar con qué color de fondo de página quiere que aparezca cada vez que ingresa al sitio. Al color seleccionado por el visitante lo almacenaremos en una cookie. En caso que no exista el color, por defecto es blanco.

La primera página mostrará un formulario con tres controles de tipo radio para la selección del color. También esta página verificará si existe la cookie creada, en caso afirmativo fijará el fondo de la página con el valor de la cookie. Tengamos en cuenta que la primera vez que ejecutemos este programa la página es de color blanco, luego variará

57_alumnos15.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php if (isset($_COOKIE['color'])) echo " bgcolor=\"$_COOKIE[color]\""
?>
>
<form action="pagina2.php" method="post">
Seleccione de que color desea que sea la página de ahora en más:<br>
<input type="radio" value="rojo" name="radio">Rojo<br>
<input type="radio" value="verde" name="radio">Verde<br>
<input type="radio" value="azul" name="radio">Azul<br>
<input type="submit" value="Crear cookie">
</form>
</body>
</html>
```

pagina2.php

```
<?php
if ($_REQUEST['radio']=="rojo")
    setcookie("color","#ff0000",time()+60*60*24*365,"/");
elseif ($_REQUEST['radio']=="verde")
    setcookie("color","#00ff00",time()+60*60*24*365,"/");
elseif ($_REQUEST['radio']=="azul")
    setcookie("color","#0000ff",time()+60*60*24*365,"/");
?>
<html>
<head>
<title>Problema</title>
</head>
<body>
Se creó la cookie.
<br>
<a href="pagina1.php">Ir a la otra página</a>
</body>
</html>
```

PRACTICA 34 – Creacion y lectura de una cookie

Crear una cookie que almacene el nombre del visitante en el sitio y cada vez que ingresemos al sitio nos de la bienvenida imprimiendo nuestro nombre. Para cargar el nombre, crear un formulario con un control de tipo text.

Ejercicio 58 – Borrado de cookie

Para borrar una cookie se debe llamar a la funcion setcookie con una fecha anterior a la actual. Haremos un algoritmo muy comun a muchos sitios que administran webmail. Recordaremos en una cookie el mail ingresado por el operador.

58_alumnos16.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese su mail:
<input type="text" name="mailusuario"
value="<?php if (isset($_COOKIE['mail'])) echo $_COOKIE['mail'];?>">
<br>
<input type="radio" name="opcion" value="recordar">
Recordar en esta computadora el mail ingresado.
<br>
<input type="radio" name="opcion" value="norecordar">
No recordar.
<br>
<input type="submit" value="confirmar">
</form>
</body>
</html>
```

pagina2.php

```
<?php
if ($_REQUEST['opcion']=="recordar")
    setcookie("mail",$_REQUEST['mailusuario'],time()+(60*60*24*365),"/");
elseif ($_REQUEST['opcion']=="norecordar")
    setcookie("mail","",time()-1000,"/");
?>

<html>
<head>
<title>Problema</title>
</head>
<body>
```



```

<?php
if ($_REQUEST['opcion']=="recordar")
    echo "cookie creada";
elseif ($_REQUEST['opcion']=="norecordar")
    echo "cookie eliminada";
?>
<br>
<a href="pagina1.php">Ir a la otra página</a>
</body>
</html>

```

PRACTICA 35 – Borrado de una cookie

Confeccionar una pagina que simule ser la de un periodico. La misma debe permitir configurar que tipo de titular deseamos que aparezca al visitarla, pudiendo ser:

Noticia politica.

Noticia economica.

Noticia deportiva.

Mediante tres objetos de tipo radio, permitir seleccionar que titular debe mostrar el periodico.

Almacenar en una cookie el tipo de titutal que desea ver el cliente. La primera vez que visita el sitio deben aparecer los tres titulares.

Ejercicio 59 – Cookie de sesion

NOTA: Para crear una cookie que sólo tenga existencia mientras no cerremos la ventana del navegador, pasaremos como fecha de expiración de la cookie, el valor cero.

Una vez que la instancia del navegador se cierra, dicha cookie desaparecerá.

Este tipo de cookie puede ser muy útil para validar un usuario en un conjunto de páginas, si previamente ingresó correctamente su nombre de usuario y clave. Es decir, una vez validado el usuario, se verifica en páginas sucesivas si existe la cookie. Una vez que el usuario cierra el navegador, no hay posibilidad de solicitar las páginas recorridas sin previa validación nuevamente de clave y usuario.

Entonces la sintaxis es:

setcookie(,0)

Veamos un pequeño ejemplo para crear y verificar si existe una cookie de sesión.

59_alumnos17.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>

```

```

<?php
if (isset($_COOKIE['usuario']))
echo "Cookie de sesión creada. Su valor es $_COOKIE[usuario]";
else
echo "No existe cookie de sesión";
?>
<br>
<a href="pagina2.php">Crear cookie de sesión</a>
</body>
</html>

```

pagina2.php

```

<?php
setcookie("usuario","diego",0);
?>
<html>
<head>
<title>Problema</title>
</head>
<body>
Cookie de sesión creada.<br>
<a href="pagina1.php">Retornar a la página anterior.</a>
</body>
</html>

```

Ejercicio 60 – Variables de sesión

NOTA: Es otro método para hacer que variables estén disponibles en múltiples páginas sin tener que pasarlas como parámetro. A diferencia de las cookies, las variables de sesión se almacenan en el servidor y tienen un tiempo limitado de existencia.

Para identificar al usuario que generó las variables de sesión, el servidor genera una clave única que es enviada al navegador y almacenada en una cookie. Luego, cada vez que el navegador solicita otra página al mismo sitio, envía esta cookie (clave única) con la cual el servidor identifica de qué navegador proviene la petición y puede rescatar de un archivo de texto las variables de sesión que se han creado. Cuando han pasado 20 minutos sin peticiones por parte de un cliente (navegador) las variables de sesión son eliminadas automáticamente (se puede configurar el entorno de PHP para variar este tiempo).

Una variable de sesión es más segura que una cookie ya que se almacena en el servidor. Otra ventaja es que no tiene que estar enviándose continuamente como sucede con las cookies. Otra ventaja de emplear una variable de sesión en lugar de una cookie es que cuando el navegador del cliente está configurado para desactivar las cookies las variables de sesión, tienen forma de funcionar (enviando la clave como parámetro en cada hipervínculo).

Como desventaja podemos decir que ocupa espacio en el servidor.

Haremos un problema muy sencillo, cargaremos en un formulario el nombre de usuario y clave de un cliente, en la segunda página crearemos dos variables de sesión y en una tercera página recuperaremos los valores almacenados en las variables de sesión.

60_varsesion.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese nombre de usuario:
<input type="text" name="campousuario"><br>
Ingrese clave:
<input type="password" name="campoclave"><br>
<input type="submit" value="confirmar">
</form>
</body>
</html>
```

pagina2.php

```
<?php
session_start();
$_SESSION['usuario']=$_REQUEST['campousuario'];
$_SESSION['clave']=$_REQUEST['campoclave'];
?>
<html>
<head>
<title>Problema</title>
</head>
<body>
Se almacenaron dos variables de sesión.<br><br>
<a href="pagina3.php">Ir a la tercer página donde se recuperarán
las variables de sesión</a>
</body>
</html>
```

pagina3.php

```
<?php
session_start();
?>
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
echo "Nombre de usuario recuperado de la variable de sesión:".$_SESSION['usuario'];
echo "<br><br>";
echo "La clave recuperada de la variable de sesión:".$_SESSION['clave'];
?>
</body>
</html>
```

PRACTICA 36 – Variables de sesion (\$_SESSION)

Confeccionar un formulario que solicite ingresar el mail de un alumno. Si el mail existe en la tabla ALUMNOS, rescatar su nombre y almacenarlo en una variable de sesion. Ademas disponer un hipervinculo a una tercera pagina que verifique si existe la variable de sesion y de la bienvenida al alumno, en caso contrario mostrar un mensaje indicando que no puede visitar esta pagina (para saber si una variable de sesion esta definida llamamos a la funcion isset).

If (isset(\$_SESSION['nombre']))

Ejercicio 61 – Incluir un archivo externo

NOTA: Hasta hora hemos visto que un archivo contiene todo el código (HTML y PHP), pero el lenguaje PHP nos permite crear librerías de funciones u objetos que veremos más adelante.

La idea fundamental de las librerías es agrupar funciones comunes a muchas páginas, no tener que tipearlas en cada archivo, lo que supone que cuando haya que hacer cambios en esas funciones las debemos localizar y modificar y dicha modificación afectará a todos los archivos donde se las utiliza.

La implementación de librerías nos permite hacer que un sistema sea más modular y facilita su mantenimiento.

Para probar esta característica del lenguaje, implementaremos dos funciones en la página "pagina2.php" y llamaremos a dichas funciones en la "página1.php"

61_incluirarchivo.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php require_once("pagina2.php");
cabeceraPagina("Titulo principal de la página");
echo "<br><br><center>Este es el cuerpo de la página<br><br></center>";
piePagina("Pie de la página");
?>
</body>
</html>
```

pagina2.php

```
<?php
function cabeceraPagina($tit)
{
    echo "<table width=\"100%\"><tr><td bgcolor=\"#ffff00\"
        align=\"center\">$tit</td></tr></table>";
}
```

```
function piePagina($tit)
{
    echo "<table width=\"50%\" align=\"center\"><tr><td
    bgcolor=\"#cccccc\">$tit</td></tr></table>";
}
?>
```

PRACTICA 37 – Incluir un archivo externo (require_once)

Confeccionar una librería que contenga una función llamada retornarConexion, la misma debe llamar a las funciones mysql_connect y mysql_select_db y retornar la variable que genere la función mysql_connect.

Tener en cuenta que a esta función la deben implementar en el archivo “pagina2.php”. En el archivo “pagina1.php” incluir la librería que contiene la función retornarConexion. Luego imprimir todos los registros de la tabla ALUMNOS.

Ejercicio 62 – Redireccionar a otra pagina

NOTA: Con PHP se puede implementar un pequeño programa que permita que cuando un usuario (navegador) solicita una página, la misma redirija a otra página.

Para probar el algoritmo implementaremos un formulario que solicite el ingreso por teclado de una dirección de internet. La segunda página redireccionará al sitio web cargado en el formulario. Es decir la segunda página nunca llega al cliente, sino que redirecciona a otra página.

La llamada a la función header debe hacerse antes de cualquier salida HTML, sino, no funcionará. Debemos pasarle como parámetro un string con el texto Location y la dirección del sitio y/o página a recuperar.

62_redireccion.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese dirección de sitio web (ej www.google.com):
<input type="text" name="direccion" size="30"><br>
<input type="submit" value="Redireccionar">
</form>
</body>
</html>
```

pagina2.php

```
<?php
header("Location: http://$_REQUEST[direccion]");
?>
```

PRACTICA 38 – Redireccionar a otra pagina (header)

Confeccionar un programa que solicite el ingreso de una clave en un formulario. La segunda página debe verificar si ingresó el string "z80" y mostrar un mensaje de bienvenida, en caso contrario, esta página debe redireccionar a la primera página nuevamente.

Cuando tenemos que redireccionar a una página que está en el mismo sitio, sólo es necesario disponer su nombre:

```
<?php
if ($_REQUEST['direccion']<>"z80")
header("Location: pagina1.php");
?>
```

Otra cosa interesante que podemos hacer es pasar como parámetro en la segunda página un código de error:

```
<?php
if ($_REQUEST['direccion']<>"z80")
header("Location: pagina1.php?error=1");
?>
```

Y luego en la primera página mostrar un mensaje de error si es que la página recibe este parámetro:

```
<?php
if (isset($_REQUEST['error']))
echo "Ingreso clave incorrecta";
?>
```

Ejercicio 63 – Creacion de imágenes dinamicas

NOTA: Existe un conjunto de funciones que nos permite la creación de un archivo de imagen (jpg, png, gif, etc.) en el servidor y posterior envío al navegador que la solicitó. Es decir que, con PHP, no sólo podemos hacer páginas dinámicas sino también imágenes dinámicas.

Veamos un ejemplo sencillo y útil donde aplicar la generación de una imagen dinámica. Casi todos hemos visto que los sitios nos obligan a ingresar un código verificador para registrarnos a un servicio de internet. Generalmente es un gráfico con una serie de números y letras poco legible (esto se hace para confirmar que quien está ingresando los datos se trata de un ser humano y no un programa de computadora camuflado como persona).

Este tipo de problema se adapta muy bien para emplear la creación de imágenes dinamicas.

63_imagendinamica.php

```

<?php
$ancho=100;
$alto=30;
$imagen=imageCreate($ancho,$alto);
$amarillo=ImageColorAllocate($imagen,255,255,0);
ImageFill($imagen,0,0,$amarillo);
$rojo=ImageColorAllocate($imagen,255,0,0);
$valoraleatorio=rand(100000,999999);
ImageString($imagen,5,25,5,$valoraleatorio,$rojo);
for($c=0;$c<=5;$c++)
{
    $x1=rand(0,$ancho);
    $y1=rand(0,$alto);
    $x2=rand(0,$ancho);
    $y2=rand(0,$alto);
    ImageLine($imagen,$x1,$y1,$x2,$y2,$rojo);
}
Header ("Content-type: image/jpeg");
ImageJPEG ($imagen);
ImageDestroy($imagen);
?>

```

PRACTICA 39 – Creacion de imágenes dinamicas desde PHP

Confeccionar una imagen dinámica que represente un botón. Utilizar las funciones vistas para imprimir el texto del botón y para rellenar regiones del gráfico, emplear la función `imageFilledRectangle`.

`ImageFilledRectangle` crea un rectángulo relleno con color "col" en la imagen "imagen", comenzando con la coordenada superior izquierda (x1, y1) y finalizando en la coordenada inferior derecha (x2, y2).

`imagefilledrectangle (imagen,x1,y1,x2,y2,col)`

Recordar que debe ser PHP puro, por lo que deberá borrar todo el código HTML que se encuentra fuera de las marcas:

```
<?php ?>
```

Ejercicio 64 – Captcha en HTML

NOTA: En un concepto anterior habíamos visto como crear un archivo gráfico en forma dinámica y el envío del mismo al navegador que lo solicitó. Ahora veremos como crear el archivo de la imagen y su posterior incorporación en una página HTML.

Continuaremos con el problema del dígito de verificación, pero ahora lo incorporaremos en un formulario donde el operador debe tipear el número que está viendo. En una tercera página verificaremos si ingresó el mismo valor que se generó en forma aleatoria.

64_captcha2.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina3.php" method="post">
Dígitos verificadores:
<br>
Ingrese valor:
<input type="text" name="numero">
<br>
<input type="submit" value="Verificar">
</form>
</body>
</html>
```

pagina2.php

```
<?php
$ancho=100;
$alto=30;
$imagen=imageCreate($ancho,$alto);
$amarillo=ImageColorAllocate($imagen,255,255,0);
ImageFill($imagen,0,0,$amarillo);
$rojo=ImageColorAllocate($imagen,255,0,0);
$valoraleatorio=rand(100000,999999);
session_start();
$_SESSION['numeroaleatorio']=$valoraleatorio;
ImageString($imagen,5,25,5,$valoraleatorio,$rojo);
for($c=0;$c<=5;$c++)
{
    $x1=rand(0,$ancho);
    $y1=rand(0,$alto);
    $x2=rand(0,$ancho);
    $y2=rand(0,$alto);
    ImageLine($imagen,$x1,$y1,$x2,$y2,$rojo);
}
Header ("Content-type: image/jpeg");
ImageJPEG ($imagen);
ImageDestroy($imagen);
?>
```


pagain3.php

```

<?php
session_start();
?>
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
if ($_SESSION['numeroaleatorio']==$_REQUEST['numero'])
    echo "Ingresó el valor correcto";
else
    echo "Incorrecto";
?>
</body>
</html>

```

PRACTICA 40 – Agregar imágenes dinamicas en un archivo HTML

Confeccionar un formulario que pida ingresar la dirección de un sitio de internet y mediante un control de tipo select permita dar un puntaje a la misma (cargar los valores de 0 a 5). Luego, en la segunda página, imprimir el nombre del sitio y un gráfico creado en forma dinámica con el puntaje obtenido (hacer un círculo por cada punto).

Un poco de ayuda para este problema:

En el archivo "pagina2.php" debemos disponer la marca HTML img para mostrar la imagen y debemos pasar a este archivo el valor que nos llegó del formulario, es decir, el puntaje seleccionado:

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
echo "La direccion: $_REQUEST[direccion] tiene";
?>

</body>
</html>

```

Por último, queda como actividad, dibujar tantos círculos como indica el parámetro puntos: Recordemos que para recuperar el parámetro puntos lo accedemos mediante el vector asociativo `$_REQUEST`:

```
$_REQUEST['puntos']
```

Para dibujar los círculos rellenos debemos emplear la función:

```
imagefilledellipse ( <manejador de imagen>, <centro en x>,
                    <centro en y>, <ancho del elipse>,
                    <alto del elipse>,<manejador de color>)
```

Ejercicio 65 – Fechas

NOTA: La funcion date retorna un string con una fecha y hora, o partes de ella según un string de formato que le pasamos como parametros. Se obtiene la fecha y hora del servidor.

65_fecha1.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
echo "La fecha de hoy es:";
$fecha=date("d/m/Y");
echo $fecha;
echo "<br>";
echo "La hora actual es:";
$hora=date("H:i:s");
echo $hora;
echo "<br>";
?>
<a href="pagina2.php">Siguiete problema</a>
</body>
</html>
```

pagina2.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
echo "La fecha de hoy es:";
$fecha=date("j/n/y");
echo $fecha;
echo "<br>";
?>
<a href="pagina3.php">Siguiete problema</a>
</body>
</html>
```

pagina3.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$dato=date("L");
if ($dato==1)
    echo "Año bisiesto";
else
    echo "Año no bisiesto";
echo "<br>";
echo "Día de la semana:";
$dato=date("w");
switch ($dato) {
    case 0: echo "domingo";
            break;
    case 1: echo "lunes";
            break;
    case 2: echo "martes";
            break;
    case 3: echo "miércoles";
            break;
    case 4: echo "jueves";
            break;
    case 5: echo "viernes";
            break;
    case 6: echo "sábado";
            break;
}
?>
</body>
</html>
```

PRACTICA 41 – Administracion de fechas y horas (funcion date)

Confeccionar un programa que permita registrar una queja en un archivo de texto. Disponer un formulario que permita almacenar el nombre del visitante (objeto HTML de tipo "text"), la descripción de la queja (objeto de tipo "textarea").

Almacenar en el archivo de texto la fecha y hora de la queja.

El archivo "pagina1.php" debe ser el formulario de ingreso de datos.

El archivo "pagina2.php" debe registrar la información en el archivo de datos.

Por último el archivo "pagina3.php" debe imprimir el archivo con todas las quejas registradas hasta el momento. Disponer un hipervínculo en el archivo "pagina1.php" para poder ver todas las quejas registradas.

Recordar que el archivo de texto obligatoriamente debe llamarse: "datos.txt".

Ejercicio 66 – Validacion de una fecha ingresada

NOTA: Si disponemos en forma separada del día, mes y año hay, una función que nos indica si se trata de una fecha válida:

```
boolean checkdate ( mes, dia, año)
```

66_checkdate.html

```
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese una fecha (dd/mm/aaaa):
<input type="text" name="dia" size="2">
<input type="text" name="mes" size="2">
<input type="text" name="anio" size="4">
<br>
<input type="submit" value="validar">
</form>
</body>
</html>
```

66_checkdate1.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
if (checkdate($_REQUEST['mes'],$_REQUEST['dia'],$_REQUEST['anio']))
    echo "La fecha ingresada es correcta";
else
    echo "La fecha no es válida";
?>
</body>
</html>
```

66_checkdate2.php

```
<html>
<head>
<title>Problema</title>
</head>
```

```

<body>
<?php
if (is_numeric($_REQUEST['dia']) &&
    is_numeric($_REQUEST['mes']) &&
    is_numeric($_REQUEST['anio']))
{
    if (checkdate($_REQUEST['mes'], $_REQUEST['dia'], $_REQUEST['anio']))
        echo "La fecha ingresada es correcta";
    else
        echo "La fecha no es válida";
}
else
    echo "La fecha no es válida";
?>
</body>
</html>

```

PRACTICA 42 – Validacion de una fecha ingresada por teclado (checkdate)

Confeccionar un formulario que solicite la carga de una fecha, disponer tres controles HTML de tipo “select” para elegir el día, el mes y el año. Validar si se trata de una fecha valida en la segunda pagina.

Ejercicio 67 – Carga de una fecha en una tabla

NOTA: La tabla alumnos tiene un campo que no habíamos nombrado llamado "fechanac" que es de tipo "date", es decir que permite almacenar una fecha. "fechanac" almacenará la fecha de nacimiento del alumno.

Veremos cual es la estructura que debemos darle a la fecha para que el MySQL la tome como válida. Haremos el alta de la tabla alumnos que habíamos visto anteriormente añadiéndole la carga de la fecha de nacimiento.

67_fechas.html

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<form action="pagina2.php" method="post">
Ingrese nombre:
<input type="text" name="nombre"><br>
Ingrese mail:
<input type="text" name="mail"><br>
Ingrese la fecha de nacimiento (dd/mm/aaaa):

```

```

<input type="text" name="dia" size="2">
<input type="text" name="mes" size="2">
<input type="text" name="anio" size="4">
<br>
Seleccione el curso:
<select name="codigocurso">
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select codigo,nombrecur from cursos",$conexion) or
    die("Problemas en el select:".mysql_error());
while ($reg=mysql_fetch_array($registros))
{
    echo "<option value=\"".$reg[codigo]\">".$reg[nombrecur]</option>";
}
?>
</select>
<br>
<input type="submit" value="Registrar">
</form>
</body>
</html>

```

67_fecha1.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la seleccion de la base de datos");
$fechanacimiento=$_REQUEST['anio']."-".$_REQUEST['mes']."-".$_REQUEST['dia'];
mysql_query("insert into alumnos(nombre,mail,codigocurso,fechanac) values
    ('".$_REQUEST[nombre].','.$_REQUEST[mail].',
    $_REQUEST[codigocurso],'$fechanacimiento')", $conexion) or
    die("Problemas en el select".mysql_error());
mysql_close($conexion);
echo "El alumno fue dado de alta.";
?>
<br>
<a href="pagina3.php">ver listado de alumnos</a>
</body>
</html>

```

67_fecha2.php

```

<html>
<head>
<title>Problema</title>
</head>
<body>

```

```

<?php
$conexion=mysql_connect("localhost","root","z80") or
    die("Problemas en la conexion");
mysql_select_db("phpfacil",$conexion) or
    die("Problemas en la selección de la base de datos");
$registros=mysql_query("select alu.codigo as codigo,nombre,mail,
                        codigocurso,fechanac,nombrecur from alumnos as alu
                        inner join cursos as cur on cur.codigo=alu.codigocurso",
                        $conexion) or
    die("Problemas en el select:".mysql_error());
while ($reg=mysql_fetch_array($registros))
{
    echo "Codigo:".$reg['codigo']."<br>";
    echo "Nombre:".$reg['nombre']."<br>";
    echo "Mail:".$reg['mail']."<br>";
    echo "Fecha de Nacimiento:".$reg['fechanac']."<br>";
    echo "Curso:".$reg['nombrecur']."<br>";
    echo "<hr>";
}
mysql_close($conexion);
?>
</body>
</html>

```

PRACTICA 43 – Carga de una fecha en una tabla de MySQL

Confeccionar un programa que permita efectuar el alta en la tabla de alumnos. Para el ingreso de la fecha de nacimiento del alumno, disponer 3 controles de tipo "select" (en el día disponer valores entre 1 y 31, en el mes valores comprendidos entre 1 y 12 y por último en el año disponer valores comprendidos entre 1900 y 2006)

Validar la fecha en la página que se efectúa el insert propiamente dicho.

Recomendación: Plantear una estructura repetitiva de PHP para la creación del control "select" donde se selecciona el año de nacimiento.

Ejercicio 68 – Funcion printf()

NOTA: Hasta ahora siempre hemos impreso dentro de la página, utilizando el comando echo, pero en ocasiones que necesitamos mayor control sobre el formato de impresión, podemos utilizar la función printf.

La función printf requiere como primer parámetro una cadena de control donde se indica cómo deben imprimirse el resto de parámetros de la misma función.

El siguiente ejemplo muestra el contenido de una variable entera con distintos formatos, lo mismo hacemos para una variable de tipo double.

68_printf1.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$entero=255;
printf("Valor entero en formato decimal %d <br>", $entero);
printf("Valor entero en formato hexadecimal con letras minúsculas
%x<br>", $entero);
printf("Valor entero en formato hexadecimal con letras mayúsculas
%X<br>", $entero);
printf("Valor entero en formato binario %b<br>", $entero);
printf("Valor entero en formato octal %o<br>", $entero);
$letra=65;
printf("Valor entero como caracter ascii %c<br>", $letra);
echo "<br>";
$real=10.776;
printf("Impresion de un valor de tipo double %f <br>", $real);
printf("Impresion de un valor de tipo double indicando la cantidad de decimales
a imprimir %0.2f <br>", $real);
?>
<br>
<A href="pagina2.php">Algunas utilidades de estas conversiones</A>
</body>
</html>
```

68_printf2.php

```
<html>
<head>
<title>Problema</title>
</head>
<body bgcolor="<?php printf("#%X%X%X",150,150,0); ?>">
En esta página definimos el color de fondo indicando la cantidad de rojo, verde
y azul, en formato decimal y solicitando a la función printf que haga la
conversión a hexadecimal. Recordemos que la propiedad bgcolor de la
marca body, lo requiere en hexadecimal.<br><br>
<a href="pagina3.php">último ejemplo</a>
</body>
</html>
```

68_printf3.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
$dia=6;
```



```
$mes=5;
$anio=2006;
printf("%02d/%02d/%d", $dia, $mes, $anio);
?>
</body>
</html>
```

PRACTICA 44 – Formateo de datos en una pagina (printf)

Confeccionar un formulario que solicite la carga del nombre de una persona, su mail y un importe que desea donar. En la pagina que procesa los datos ingresados, mostrar el importe a donar rellenando con ceros, por ejemplo: \$ 0000170 dolares.

Ejercicio 69 – Funcion sprintf()

NOTA:

Así como vimos que la función printf nos permite tener un control más fino sobre cómo un dato debe imprimirse en una página, la función sprintf nos permite formatear la salida de un dato hacia un string y no a la página HTML.

Tipos de conversión utilizadas por la función sprintf (tener en cuenta que son los mismos caracteres de control que se aplican a la función printf):

%b Formatea un entero como un número binario.

%d Formatea un entero como un número decimal con signo.

%u Formatea un entero como un número decimal sin signo.

%o Formatea un entero como un número octal.

%x Formatea un entero como un número hexadecimal en minúsculas.

%X Formatea un entero como un número hexadecimal en mayúsculas.

%c Formatea un entero como un caracter ASCII.

%f Formatea un double con una cantidad de decimales.

%s Formatear un string.

69_sprintf.php

```
<html>
<head>
<title>Problema</title>
</head>
<body>
<?php
function retornarColorHexa($rojo,$verde,$azul)
{
    $color=sprintf("#%02X%02X%02X", $rojo, $verde, $azul);
```

```
    return $color;
}
?>
<table>
<tr>
<td bgcolor="<?php echo retornarColorHexa(255,0,0)?>">Cuadro
rojo</td>
<td bgcolor="<?php echo retornarColorHexa(0,255,0)?>">Cuadro
verde</td>
<td bgcolor="<?php echo retornarColorHexa(0,0,255)?>">Cuadro
azul</td>
</tr>
</table>
</body>
</html>
```

PRACTICA 45 – Formateo de datos y salida a un string (sprintf)

Imprimir la tabla de caracteres ASCII. Implementar una funcion que le enviemos un entero y nos retorne el carácter ASCII de dicho valor.