

EJERCICIOS de php. Relación larga.

1.- Elabora también una página que mezcle PHP con HTML, la página debe mostrar por pantalla lo siguiente:

- a) “Esta es la primera práctica escrita en PHP” en código PHP.
- b) “Escrita por Apellidos y Nombre del alumno/a” en código HTML.

2.- Sobre el código PHP del ejercicio anterior añade dos comentarios:

- a) El primero justo al principio del código PHP (después de <?) indicando que es el principio del código PHP.
- b) El segundo justo antes de finalizar el código PHP (antes de ?>, indicando que es el final del código PHP.

3.- Partiendo de 2 variables (\$primera y \$segunda) con valores 4 y 8, respectivamente, hacer una página PHP que calcule:

- a) La suma de \$primera y \$segunda.
- b) El producto de \$primera y \$segunda.

4.- Partiendo de 2 variables (\$primera y \$segunda) con valores 10 y 2, respectivamente, hacer una página PHP que calcule:

- a) La diferencia de \$primera menos \$segunda.
- b) La división de \$primera entre \$segunda.

5.- Partiendo de 2 variables (\$primera y \$segunda) con valores 4 y 8, respectivamente, hacer una página PHP que calcule:

- a) \$primera elevada a \$segunda.
- b) Raíz cuadrada de \$primera.
- c) Raíz cuadrada de \$segunda.

6.- Tenemos dos cadenas \$cadena1 con valor “hola a todo el mundo” y \$cadena2 con valor “mi nombre es nombre y apellidos del alumno/a”. Se pide:

- a) \$cadena3 contendrá el valor de la concatenación de \$cadena1 y \$cadena2, mostrar por pantalla el contenido de \$cadena3.
- b) \$cadena1 contendrá el resultado de la concatenación de sí misma con \$cadena2, mostrar por pantalla el contenido de \$cadena1.

7.- Hacer una página PHP que compruebe si el contenido de una variable es PAR o IMPAR.

8.- Hacer una página PHP que tenga dos variables y nos indique si la suma de los números es mayor que el producto de ellos o al contrario. Ej.:

\$numero1=1; \$numero2=2 ----> “la suma es mayor que el producto”

\$numero1=3; \$numero2=2 ----> “el producto es mayor que la suma”

9.- Hacer una página PHP que tenga dos variables y según el valor de dichas variables realice los siguientes apartados:

- a) Si el valor de la primera variable es mayor que 0 y el valor de la segunda menor que la primera, mostrará el nombre del alumno/a.
- b) Si el valor de la primera variable es mayor que 0 y el valor de la segunda es mayor o igual que la primera, mostrará los apellidos del alumno/a.
- c) Si el valor de la primera variable es menor que 0, mostrará el nombre y apellidos completos del alumno/a, independientemente del valor de la segunda variable.

- 10.-** Hacer una página PHP que para una nota almacenada en una variable, muestre por pantalla lo siguiente:
- a) Si la nota es menor que 5 → “suspense”.
 - b) Si la nota está entre 5 y 6 → “aprobado”.
 - c) Si la nota está entre 6 y 7 → “bien”.
 - d) Si la nota está entre 7 y 8 → “notable”.
 - e) Si la nota es mayor que 8 → “sobresaliente”.
- 11.-** Hacer un programa en PHP que muestre por pantalla los números del 1 al 100 separados por comas.
- 12.-** Hacer una página PHP que muestre por pantalla los números naturales pares menores de 100 en orden descendente. Es decir: 100, 98, 96, 94..., 2, 0.
- 13.-** Hacer una página PHP que muestre por pantalla la tabla de multiplicar de una variable.
- 14.-** Hacer en PHP una página que saque la tabla de multiplicar completa del 1 al 10.
- 15.-** Hacer una página PHP que muestre una figura similar a la siguiente. Solo está permitido el uso de un * en todo el código.
- ```
*
**


```
- 16.-** Hacer una página PHP que muestre por pantalla todos los divisores (aquellos cuyo resto de la división es 0) de un número guardado en una variable. Ej.: 16 -> sus divisores son: 1, 2, 4, 8, 16.
- 17.-** Hacer en PHP una página que muestre las potencias de 2 hasta la potencia 16, es decir de  $2^1$  a  $2^{16}$ .
- 18.-** Un número primo es aquel que solo es divisible entre 1 y entre sí mismo. Hacer una página PHP que muestre por pantalla si un número es o no primo.
- 19.-** Hacer en PHP una página que muestre la fecha.
- 20.-** Hacer una página PHP que muestre los 20 primeros términos de la sucesión de Fibonacci. La sucesión de Fibonacci se caracteriza por tener el primer elemento: 1, el segundo elemento: 1 y el resto elementos: se forman sumando los 2 términos anteriores. -> 1, 1, 2, 3, 5, 8 ...
- 21.-** Hacer una página PHP que contenga una función llamada primo que reciba un número como parámetro y devuelva 1 si es primo y 0 en caso contrario.  
Usar esa función para mostrar los 30 primeros números primos.
- 22.-** Hacer una página PHP que contenga una función media que recibe 4 números como parámetros y devuelve el valor medio de los 4 parámetros recibidos.
- 23.-** Hacer una página PHP que contenga una función cuentavocales que recibe una cadena de texto y muestra por pantalla el número de vocales totales que tiene la cadena (recordad que una cadena de texto se comporta como un *array* de caracteres).

**24.-** Hacer una página PHP que contenga una función cuadrado que recibe 2 parámetros: un carácter (que puede ser cualquiera) y un número; la función debe mostrar por pantalla un cuadrado con el carácter recibido (tantas filas y columnas como indique el número).

Ejemplo: cuadrado("#",3)

```
###
```

```
###
```

```
###
```

**25.-** Hacer una página PHP que contenga una función pirámide que recibe 2 parámetros: un carácter (que puede ser cualquiera) y un número; la función debe mostrar por pantalla una pirámide invertida con el carácter recibido (tantas filas y columnas como indique el número, y de ahí descendiendo hasta 1).

Ejemplo: pirámide("@",4)

```
@@@@
```

```
@@@
```

```
@@
```

```
@
```

**26.-** Hacer una página PHP que contenga una función tiempo que recibe 2 parámetros: el tiempo que hizo ayer y el tiempo que hace hoy. Hay 3 posibilidades: SOL, NUBLADO, LLUVIA. Y en función del tiempo de ayer y el tiempo de hoy muestre la predicción para mañana. Esta predicción se basa en la siguiente tabla.

AYER, HOY → MAÑANA

SOL, SOL → SOL

NUBLADO, NUBLADO → NUBLADO

LLUVIA, LLUVIA → LLUVIA

SOL, NUBLADO → NUBLADO

NUBLADO, SOL → SOL

SOL, LLUVIA → NUBLADO

LLUVIA, SOL → NUBLADO

NUBLADO, LLUVIA → LLUVIA

LLUVIA, NUBLADO → NUBLADO

**27.-** Se pide realizar un generador para una apuesta del juego de Euromillones. Para una apuesta hay que obtener al azar cinco números principales de una lista de 50 (del 1 al 50) y además otros dos números de una lista de nueve (del 1 al 9) llamados “estrellas”.

**28.-** Crear un fichero llamado *funciones.inc.php* que incluya dentro de él las siguientes funciones:

La función/nes creada en las prácticas 22, 23, 24 y 25.

**29.-** Crear un fichero llamado *pruebafunciones1.php* que incluya dentro de él una llamada con `include()` a la actividad anterior, comprobando cada una de las funciones.

**30.-** Crear un fichero llamado *pruebafunciones2.php* que incluya dentro de él una llamada con `require()` al ejercicio 28, comprobando cada una de las funciones.

**31.-** Crear un fichero llamado *pruebafunciones3.php* que incluya dentro de él una llamada con `include_once()` al ejercicio 28, comprobando cada una de las funciones.

**32.-** Crear un fichero llamado *pruebafunciones4.php* que incluya dentro de él una llamada con `require_once()` al ejercicio 28, comprobando cada una de las funciones.

**33.-** Hacer un formulario PHP que pida una serie de datos: nombre, apellidos, dirección, teléfono, población y provincia. Este formulario enviará los datos a otra página PHP mediante el método GET.

La página que recibe los datos debe mostrar los datos recibidos por pantalla.

**34.-** Hacer un formulario PHP que pida una serie de datos: nombre, apellidos, dirección, teléfono, población y provincia. Este formulario enviará los datos a otra página PHP mediante el método POST. La página que recibe los datos debe mostrar los datos recibidos por pantalla.

**35.-** Partimos de una página donde se nos pide un usuario y una contraseña. Enviando los datos por POST y comprobamos que el usuario y contraseña sean “usuario” “usuario”. Una vez hemos autenticado al usuario en el sistema, se inicia la sesión, y en ese momento nos aparecerá un enlace a otra página Web en la que deberemos comprobar que el usuario ha sido autenticado en el sistema, mostrando esta página una sección exclusiva para los usuarios registrados. Si el usuario ha accedido a esta página sin registrarse, deberá mostrar un enlace a la página principal para que se identifique correctamente.

**36.-** Partiendo del caso práctico anterior, en la segunda página (una vez identificados), añadir un enlace que nos cierre la sesión y nos proporcione un enlace para la página inicial (aquella en la que teníamos que poner el nombre de usuario y contraseña).

**37.-** Hacer una página que se comporte de manera distinta si el usuario está autenticado en el sistema o no (ha iniciado la sesión). Si el usuario no está identificado mostrará un enlace a otra página para proceder a la autenticación (mediante un usuario y una contraseña). Una vez autenticado en el sistema deberá volver a la página inicial. Si el usuario está autenticado mostrará unos enlaces de descarga.

**38.-** Completa la clase Caja con la función introducecompleta, donde no solo se introduce el contenido, sino que también se introduce el alto, ancho, largo y color.

**39.-** Complete la clase Caja con la función muestra, donde se muestran todos los elementos de la clase, es decir, el alto, ancho, largo contenido y color.

**40.-** Hacer una página PHP que sea un caso completo de la clase Caja, donde se cree un objeto de la clase Caja, se le introduzcan elementos y se muestren por pantalla dichos elementos.