EJERCICIOS PHP MEDIO

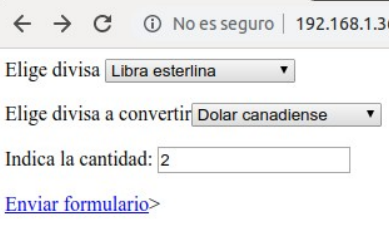
# Ejercicio 1

Crear en un único fichero php un juego de adivinar un número del 1 al 10 en tres intentos. Ha de comprobar si recibe por GET los valores "intento", "num" y “objetivo”. Si no vienen se supone que es el inicio del juego: ha de sacar un número aleatorio del 1 al 10 y en la página pedirlo mediante un formulario. En caso contrario ha de indicar cuántos intentos lleva y el mismo formulario. El formulario se envía a la misma página. Si el intento es superior a 3 ha de mostrar "Perdiste" y un enlace para volverlo a intentar. Si acierta con el número aleatorio mostrará "has ganado" y el mismo enlace. Se ha de almacenar en un input de tipo **hidden** tanto el valor de intento como el objetivo. Cada vez que se envía un número "intento" debe valer uno más.

|  |
| --- |
| **SOLUCIÓN** |
| [C:\xampp\htdocs\AdivinaNumero\index.php](file:///C:\xampp\htdocs\AdivinaNumero|index.php) |

# Ejercicio 2

Vamos a mejorar el ejercicio de conversión de divisas creando un php que muestre 4 divisas diferentes en dos desplegables, y un campo numérico. Será obligatorio para enviar el formulario que el campo numérico tenga algún valor numérico. La página del resultado muestra la conversión del campo numérico de la primera divisa a la segunda. Ha de usar un archivo externo de javascript.



|  |
| --- |
| **SOLUCIÓN** |
|  |

# Ejercicio 3

Crear un único archivo php con ciertas clases en un css. Se ha de dar a elegir mediante

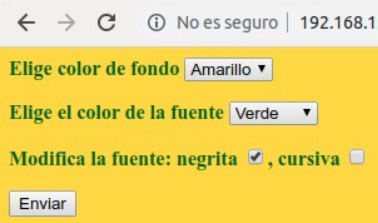
un desplegable el color de fondo, y con otro, el color de la letra. Mediante dos checkbox

se ha de dar la opción de indicar si la fuente es además cursiva o negrita. Si se envía el

formulario se carga la misma página con el cambio de estilo. Por último, las opciones

elegidas han de recordarse cuando se recargue la página. Ha de usar un archivo externo

de css.



|  |
| --- |
| **SOLUCIÓN** |
|  |

# Ejercicio 4

Crear una pequeña agenda a través de un formulario que recoja los siguientes datos de contactos: Nombre, teléfono (se debe seleccionar el tipo, fijo o móvil, a través de un control checkbox), correo electrónico y fecha de nacimiento. A través de un botón se irán añadiendo los contactos, mostrando el servidor un mensaje sobre la ejecución del proceso. En caso de producirse un error se mostrará de nuevo el formulario remarcando el campo incorrecto. Un segundo botón debe permitir resetear la información de todos los campos. Por último, otro botón permitirá visualizar en pantalla todos los contactos dados de alta en la agenda.

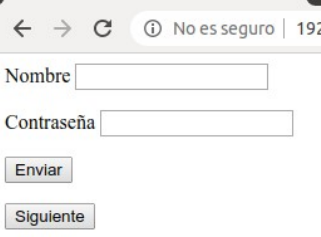
|  |
| --- |
| **SOLUCIÓN** |
|  |

# Ejercicio 5

Se pide crear cuatro páginas php. De la segunda en adelante solo se puede entrar si se

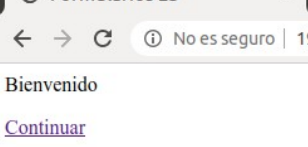
inició sesión, en caso contrario se redirige a la primera, donde se pide usuario y

contraseña.

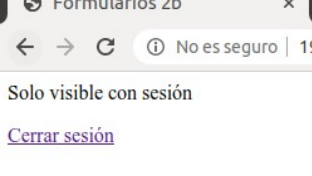


En la segunda se muestra un mensaje de bienvenida si se introdujo “profe” y “1234”. En

caso contrario se redirige a la primera página.



Habrá un enlace a la tercera.



|  |
| --- |
| **SOLUCIÓN** |
|  |

# Ejercicio 6

Crear una página php donde se de tres oportunidades para adivinar un número prefijado

como en el ejercicio 4, pero en esta ocasión usando sesiones.

|  |
| --- |
| **SOLUCIÓN** |
|  |

# Ejercicio 7

Escribe un programa PHP que permita al usuario introducir las temperaturas máximas y mínimas de las ciudades de Madrid, Barcelona, Sevilla y Bilbao (por ese orden) para cada uno de los meses del año. Se creará un array que sirva para rellenar dinámicamente con valores por defecto los elementos del formulario. Las temperaturas pueden tener valores decimales. Al enviar el formulario, el programa mostrará una tabla con los nombres de las ciudades ordenadas alfabéticamente y los datos correspondientes a la temperatura máxima, mínima y temperatura media del año en cada ciudad.

|  |
| --- |
| **SOLUCIÓN** |
|  |

# Ejercicio 8

Partiendo del ejercicio 4 donde desarrollaste una pequeña agenda, añade un par de elementos que tengan cuentas de correo no válidas, así como una función de usuario que reciba el correo electrónico de cada contacto de la agenda que devuelva true si el correo termina en “@gmail.com”, “@educa.madrid.org”, “@yahoo.com”, “@outlook.com” o “@hotmail.com” y false en caso contrario.

|  |
| --- |
| **SOLUCIÓN** |
|  |

# Ejercicio 9

Crear un formulario de login/registro con la identificación (login.php) y el registro (registro.php) para la tienda online.

El formulario de login debe tener un enlace al de registro y viceversa.

* Parte de login:

El programa tendrá un array con usuarios ya registrados que serán:

* + Usuario: admin, password: admin
  + Usuario: usuaRIO, password: u$suar10

Si un usuario se identifica correctamente como usuario normal, el programa redirigirá a una página que recibe el nombre del usuario y muestra el mensaje de ‘Bienvenido + nombre de usuario’.

Si un usuario no se identifica correctamente, en el mismo formulario se muestra un mensaje diciendo “Identificación incorrecta” y el formulario además conserva los datos que se introdujeron.

ATENCIÓN: la comparación de usuario y contraseña ha de ser exacta. Es decir, no

es lo mismo usuario que usuaRIO.

* Parte de registro:

Si una persona se registra, los datos se envían a la misma página del registro, donde se comprueba que han llegado todos los datos necesarios (los que cada uno

considere, mínimo usuario y contraseña). Si no son correctos, vuelve a mostrar el

formulario de registro indicando en color destacado los datos incorrectos o que

faltan dentro de los correspondientes campos del formulario. Si los datos son

correctos, redirigirá a login.php en cuyo formulario se mostrará el usuario y

contraseña que introdujeron en el registro.

|  |
| --- |
| **SOLUCIÓN** |
|  |

# Ejercicio 10

Partiendo del ejercicio anterior, se quiere habilitar un mecanismo para subir imágenes al servidor. Para ello:

* Si el usuario se identifica correctamente como administrador (admin), se mostrará una pantalla en la que hay un menú inicialmente solo con un elemento “Nuevo artículo”.
* Si pulsa ese enlace se redirigirá o incluirá (a tu criterio) a una página con un formulario para crear un nuevo artículo con los datos que necesites.
* Debe haber un campo que permita subir al menos una imagen para cada producto nuevo que se guardará en una carpeta del proyecto llamada “img”.
* Al enviar el formulario se mostrarán los datos introducidos y la imagen.

*NOTA: hay que entregar el directorio img vacío, y las imágenes que se van a subir en un*

*directorio separado del código que se llame fotos.*

|  |
| --- |
| **SOLUCIÓN** |
|  |

# 