

PRÁCTICA 2

1. Crea una cadena de texto con el texto "Lo que estoy escribiendo es un string". Crea una segunda cadena de texto "muuuuuy largo". Concatena las dos cadenas en una sola y luego muéstrala. Llama a una función que diga la posición de la primera 'e' y de la última 'u'. También calcula la posición de la palabra "texto".

2. Crea un array de nombres de alumnos, que contengan nombre de al menos 5 alumnos. A continuación, crea una función llamada

convierteAlumnos(\$nombres, \$opcion)

donde el primer parámetro sea el array con los nombres de los alumnos, y el segundo parámetro pueden ser tres opciones:

- "L": transforma todos los strings del array \$nombres a minúsculas y lo devuelve.
- "U": transforma todos los strings del array \$nombres a mayúsculas y lo devuelve.
- "M": transforma todos los strings del array \$nombres de modo que la primera letra de cada nombre de empresa sea mayúscula y el resto minúscula. Lo devuelve.

Muestra un ejemplo de la función con cada una de las diferentes opciones.

3. Crea una cadena llamada \$direccion_ip y asígnale una dirección ip como 192.168.11.200. A continuación, separa en un array con cada dígito de la dirección ip, y muestra cada dígito por separado. Seguidamente reconstruye en una cadena la dirección ip, pero que en lugar de separar por puntos los dígitos aparezcan separados por dos puntos (:) y muéstralo. [Pista: usar explode e implode]

4. Crear un array llamado \$word_list_en con 40 palabras en inglés. Crea otro array llamado \$word_list_es con las mismas 40 palabras en el mismo orden, pero en español. El ejercicio consiste en hacer un **traductor literal** de español a inglés o viceversa, es decir, creamos una variable con una cadena de texto, debe recorrerla y devolverla en el idioma traduciendo una por una las palabras (se supone que están en la misma posición en los arrays). Para la cadena de texto a traducir usa solo palabras de entre las 40 que has utilizado.

5. Vamos a construir un encriptador y desencriptador de mensajes. Crearemos dos funciones:

- *encriptar(\$mensaje,\$clave)*
 - o donde el primer argumento sea el mensaje a encriptar
 - o con \$clave=3 se transformará en en la f.
 - o La función devolverá el mensaje cifrado sustituyendo los espacios en blanco del final y cada letra del mensaje por la correspondiente según la clave.

- `desencriptar($mensaje,$clave)`
 - o donde el primer argumento sea el mensaje a encriptar
 - o con `$clave=3` se transformará en en la b.
 - o La función devolverá el mensaje cifrado sustituyendo cada letra del mensaje por la correspondiente según la clave.

Para mostrar que lo has hecho bien encripta un mensaje y muéstralo, desencriptalo y muestra el mensaje que coincide con el original. **Pista:** busca y utiliza las funciones PHP para pasar un carácter a su correspondiente dígito ASCII y viceversa.

6. Mejora el programa anterior de tal manera que el mensaje original lo divida primero en un array de palabras considerando el espacio en blanco como separador únicamente. A continuación, debe poner cada palabra del revés (hola -> aloh). Seguidamente encriptará cada palabra usando la función del ejercicio anterior. Finalmente devolverá un string con cada palabra encriptada añadiendo un espacio en blanco entre cada palabra. El desencriptador hará lo contrario (y no digo más). Muestra el programa funcionando encriptando y desencriptando.

7. Vamos a crear un programa que calcule el IVA de un carrito de la compra. El carrito será un array bidimensional asociativo de este tipo: (Puedes añadir más productos y más campos a tu elección)

```
$carrito = array(
    array("id" => 1234, "nombre" => "PS4", "precio" => 349.95, "cant" => 2, "iva_r" => 0),
    array("id" => 1235, "nombre" => "iPhone XS", "precio" => 1249.95, "cant" => 1, "iva_r" => 0),
    array("id" => 1236, "nombre" => "Chocolate", "precio" => 9.95, "cant" => 5, "iva_r" => 1)
)
```

Deberéis crear una función llamada `subtotal($linea_pedido)` que calcule el precio de cada línea de pedido, multiplicando el precio por la cantidad y aplicando el iva correspondiente, si `iva_r` es 0 será del 21% y si `iva_r` es 1 será del 10%.

Mostrar en una tabla HTML el carrito de la compra (nombre, precio, cantidad y subtotal). En la última fila aparecerá el total del pedido a pagar.

Se tendrá en cuenta la visualización y el estilo del carrito de la compra resultante.

8. Vamos a crear una librería online, pero con los libros almacenados en un array asociativo. Los datos a guardar por cada libro son: ISBN, título, descripción, categoría, editorial, foto, precio.
 - La categoría puede ser: ciencias, cocina, deporte, novela, historia, scifi, negra, romántica.
 - El campo foto será una url en tu pc en la misma carpeta que esté el fichero php nos crearemos una carpeta **imgs** donde meteremos las imágenes de cada libro llamándolas con el isbn.jpg. El tamaño de cada imagen será aproximadamente 100x150px.
 - ISBN será un número de trece dígitos que se puede tratar como una cadena.
 - Precio será un float con dos decimales.

A continuación, te muestro cómo debería quedar la visualización del array. Tienes que intentar que sea lo más parecido posible. Como mínimo deberás tener 15 libros, y mostrar 4 libros en cada fila. Los datos e imágenes de los libros deben ser lo más reales posible. Debes tener como mínimo 5 libros de novela histórica y 5 de novela negra, pero sólo mostraremos los 4 primeros de cada una de esas categorías.

LIBRERÍA JAROSO 2018

Novela histórica



El manuscrito de piedra (los manuscritos 1) (Tapa blanda)
LUIS GARCÍA JAMBRINA
11,40€



El manuscrito de piedra (los manuscritos 1) (Tapa blanda)
LUIS GARCÍA JAMBRINA
11,40€



El manuscrito de piedra (los manuscritos 1) (Tapa blanda)
LUIS GARCÍA JAMBRINA
11,40€



El manuscrito de piedra (los manuscritos 1) (Tapa blanda)
LUIS GARCÍA JAMBRINA
11,40€

Novela negra

Realizado por <nombre>

9. Instagram casero. Intenta hacer con diseño responsive (puedes usar Bootstrap) una web que muestre unas 7 fotos con la misma información que se muestra en esa red social:

- Nombre del usuario que hace el post
- Lugar donde se toma la foto
- Fotografía
- Número de me gusta
- Comentarios: cada uno con usuario y mensaje. Cada foto añade dos comentarios al menos.

Ten en cuenta que la información a mostrar estará en un array, y al recorrerlo lo vas pintando. Si tienes problemas para hacerlo responsive hazlo como puedas.

10. En este ejercicio tenéis que terminar el ejercicio hecho en clase sobre un instituto. Debe mostrar información sobre cursos y alumnos. La información para cada **curso** es:

- Nombre del curso (1º ESO A, 1º ESO B, 2º Bachillerato A, 1º DAW, 2º DAW, etc.)
- Etapa educativa (ESO, Bachillerato, Ciclo Medio, Ciclo Superior)
- Año académico (2020, 2021, 2022, 2023, 2024, 2025 ... 2030)

La información para cada **alumno** es:

- Apellidos
- Nombre
- Edad
- DNI
- Email
- Localidad
- Teléfono
- Curso (igual que "Nombre del curso" en Cursos)
- Avatar (una imagen en la carpeta imgs/ con el nombre el dni del alumno.jpg)

Haremos dos arrays de arrays asociativos para los cursos y para los alumnos, p. ej. para los cursos sería algo así, con los que quieras añadir:

```
$cursos = array(
    array("nombre" => "1º DAW", "etapa" => "Ciclo Superior", "curso" => 2021),
    array("nombre" => "1º GA", "etapa" => "Grado Medio", "curso" => 2021),
    array("nombre" => "1º Laboratorio", "etapa" => "Grado Medio", "curso" => 2021)
);
```

Crearemos una web con un menú lateral donde podamos elegir como opción "Alumnos" y otra "Cursos". La primera mostrará todos los alumnos y la segunda todos los cursos. Ahora no haremos nada más, en próximas prácticas le añadiremos más funcionalidad a este ejemplo, con lo cual tenéis que hacerlo bien.

Podéis usar la plantilla SB Admin 2, como he usado yo, o buscar otras plantillas (se valorará muy positivamente):

- <https://adminlte.io/>
- <https://athemes.com/collections/free-bootstrap-admin-templates/>