

1. A partir de un array de 5 valores enteros:
 - a) Ordena sus componentes utilizando la función `sort()` y `rsort()`; a continuación, inserta un nuevo valor sin perder el orden en el vector.
 - b) Busca un valor concreto, utilizando las funciones `in_array()` o `array_search()`, con y sin coincidencia de tipos.
 - c) Elimina la última componente con `unset()` o con `array_pop()`
2. A partir del array del ejercicio nº 2 de las actividades UT2_parte II (array asociativo que almacena las lectivas de los módulos de 2º de DAW) utiliza las funciones de ordenación de arrays asociativos (`asort` y `ksort`).
3. Utiliza la función `explode` e `implode` y crea:
 - a) un array asociativo que almacene los colores a partir del siguiente string: "rojo-verde-azul-amarillo".
 - b) Ejecuta `echo $persona`, una vez creado el string `$persona` a partir del siguiente array ("Pedro","González","López").
4. Determina si una frase es palíndromo. Utiliza las funciones de string para ello.
5. Traduce un texto escrito en castellano a "latín macarrónico". Para ello se cambia cada palabra del texto por otra que se construye a partir de la original poniendo la primera letra al final y terminándola en "um". Ejemplo: "una imagen vale más que mil palabras" -> "Nauum magenium alevum asmum euqum ilmum alabraspum".
6. Encripta una palabra empleando el método César, que consiste en desplazar cada carácter tres posiciones en el alfabeto.
7. Visualiza una palabra y sus rotaciones. Ejemplo: si la palabra es Hola, las rotaciones serán: aHol, laHo, olaH, Hola.
8. ¿Cuál sería el código para obtener la siguiente salida en pantalla?

Ejercicio 8

Hoy es 13-10-2022
Hoy es Thursday-October
Semana 41 del año

Y son las 12:03
Y son las 02:03 Ahora sí es correcta
Y ahora más completa 02:03:20 pm

9. Determina si a partir de una fecha para realizar una solicitud, se ha terminado el plazo.
10. Crea una función para visualizar la tabla de multiplicar de un número previamente inicializado a un valor entero. Casos:
 - a) Pasar el número como parámetro y visualizar la tabla en la función.
 - b) Pasar el número como parámetro y obtener el resultado a través de un string y la sentencia `return`.
 - c) Pasar el número como parámetro y obtener el resultado a través de un string y el paso de parámetro por referencia.
11. Poner un ejemplo de función que utiliza la asignación de valor en la cabecera (ejemplo el caso de calcular el precio con iva; si no se pasa el iva se supone que es el 21%). Los parámetros de este tipo, siempre deben estar a la derecha de cualquier otro parámetro que no tenga asignado un valor.
12. Crea una función que reciba un número entero positivo de tres cifras y devuelva el número invertido. Visualizarle en el script principal.

13. Crea una función que reciba un array asociativo que almacene las horas lectivas de los módulos de 2º DAW. La función devolverá el código del módulo con mayor horas lectivas.
14. Crea una función que, partiendo del array asociativo anterior, devuelva el array sin el módulo de mayor número de horas lectivas.
15. Crea una función que reciba una array asociativo que almacene los ingresos y gastos de una familia gallega durante un mes. El array recoge una muestra de las familias de cada provincia de esa comunidad. La función devuelve cual es la provincia que más ahorra. Visualiza la provincia.
16. Calcular la edad pasando a una función tres parámetros: el día de nacimiento, el mes y el año. Muestra la edad.
17. Crea una función para calcular cuántos días faltan para el día de navidad.
18. Muestra los fármacos que están caducados. Para ello, es necesario crear un array con 5 fármacos; de cada uno se almacena su nombre y la fecha de caducidad. Utiliza una función que reciba el array y le devuelva sin los fármacos caducados.
19. ¿Cuál es la salida de cada uno de estos scripts?:

<pre>// Caso 1 \$var1=1; function ambito1() { \$var2=\$var1; echo "Pasando por ambito1(): "; echo var_dump(\$var2)."
"; } ambito1();</pre>	<pre>//Caso 2 \$var1=1; function ambito2() { global \$var1; \$var2=\$var1; echo "Pasando por ámbito2(): "; echo \$var2."
"; } ambito2();</pre>	<pre>//Caso 3 \$var1=1; function ambito3() { global \$var1; \$var1++; echo "Pasando por ámbito3(): ", \$var1, "
"; } ambito3(); \$var1++; echo "Pasando por ambito3(): ", \$var1, "
";</pre>	<pre>// Caso 4 \$var1=1; function ambito4() { global \$var1; static \$var3=1; \$var2=\$var1+\$var3; \$var3+=10; echo "Pasando por ambito4(): "; echo \$var2." ".\$var3."
"; } ambito4(); ambito4();</pre>
<pre>//Caso 5 \$var1=1; function ambito5(\$p) { \$p++; return \$p; } echo "Pasando por ambito5(): "; echo ambito5(\$var1)."
";</pre>	<pre>//caso 6 \$var1=1; function ambito6() { global \$var1; \$GLOBALS['var2']=\$var1+10; } ambito6(); echo "Pasando por ámbito6(): \$var2
";</pre>	<pre>//Caso 7 \$var1=1; function ambito7(\$p) { \$p++; } ambito7(\$var1); echo "Pasando por ambito7(): "; echo \$var1."
";</pre>	<pre>//Caso 8 const VALOR=100; function ambito8() { \$var1=VALOR*2; define('OTRO_VALOR', -50); return \$var1; } echo "Pasando por ambito8(): " .(ambito8() + OTRO_VALOR);</pre>

20. Diseña una función para determinar si en un array de strings, se encuentra una determinada subcadena (ambos datos se pasan como parámetros. Si no se pasa la subcadena se buscará la subcadena predeterminada "ma". La función debe calcular el número de veces que aparece la subcadena entre las palabras del array.
 - a) Diseña una función de usuario convencional.
 - b) Utiliza una variable de función.
 - c) Diseña una función anónima e invoca a la función desde una función convencional (*callback*)
21. Diseña una función para que genere un número aleatorio entre 1 y 6 simulando el funcionamiento de un dado. La función deberá incluir las sentencias echo adecuada para informar del resultado obtenido; por ejemplo, si se obtiene 6 mostrar el mensaje "¡Enhorabuena!" y si no "Inténtelo de nuevo junto al valor generado".
 - a) Diseña una función de usuario convencional.
 - b) Utiliza una variable de función.

- c) Diseña una función anónima e invoca a la función desde una función convencional (*callback*)