

INSTITUTO DE E.S. “PABLO RUIZ PICASSO”. Chiclana.

CURSO: 2018/2019

Ciclo Superior: A.S.I.R.

Módulo: Implantación de Aplicaciones Web

USO DE FUNCIONES PREDEFINIDAS EN PHP. 2ª EVAL.

- Realiza un ejercicio con cada una de las funciones siguientes:

MANEJO DE CADENAS O STRING , MATRICES (ARRAYS) y FECHAS.

1.-) La función **strrev()** recibe una cadena de caracteres y la devuelve al revés.

2.-) La función **strcmp()** compara dos cadenas pasadas como argumentos sin distinguir entre letras mayúsculas y minúsculas y devuelve un número entero, que es un cero si son iguales, un número mayor que cero si la primera es mayor que la segunda y un número menor que cero si la primera es menor que la segunda.

3.-) La función **strcmpi()** compara dos cadenas pasadas como argumentos distinguiendo entre letras mayúsculas y minúsculas y devuelve los mismos valores que la función anterior con la salvedad indicada. En el mismo ejemplo anterior, las tres instrucciones siempre devolverían -1.

4.-) La función **chr()** recibe un número entero y devuelve el carácter correspondiente del código ASCII.

5.-) La función **ord()** recibe un carácter del código ASCII y devuelve el número entero correspondiente.

6.-) La función **range()** devuelve una matriz que sólo contiene los elementos indicados entre dos números, ambos incluidos. Su sintaxis es:

range(número entero del límite inferior, número entero del límite superior);

7.-) La función **checkdate()** nos permite comprobar si una fecha, que se pone como argumento, es válida. Si lo es, la función devuelve el valor True. En el caso contrario, devuelve False.

La sintaxis de esta función es: **checkdate(mes,día,año);**

8.-) La función **date()** devuelve una cadena de texto que refleja una fecha y una hora formateadas como se indique en el primer parámetro. En el segundo argumento puede indicarse la fecha que se quiere mostrar utilizando un valor de tipo timestamp (instante calculado como número de segundos transcurridos desde el 1/1/1970, época UNIX). Si no se especifica este segundo parámetro, se muestra la hora actual. La sintaxis de esta función es: **date(cadena de formato, número entero del instante o nada);**

9.-) La función **getdate()** se usa para extraer información de una fecha dada. Tampoco admite categorías locales y, por tanto, la información literal está en inglés.

La sintaxis de esta función es: **getdate(número entero del instante);**

10.-) La función **mktime()** se usa para extraer el valor timestamp de una fecha. Después, podemos utilizar este valor como argumento de otras funciones. El término timestamp se refiere al momento exacto de una fecha concreta expresado en segundos desde el día 1 de enero de 1970 (época UNIX). Esta función nos permite hallar el número de segundos que han transcurrido desde entonces hasta la fecha que especifiquemos.

11.-) La función **strftime()** se usa para formatear una fecha teniendo en cuenta las especificaciones y categorías locales. Así soslayamos los problemas de este tipo que plantean las funciones **date()** y **getdate()**.

La sintaxis de esta función es: **strftime(cadena de formato,instante);**

Si queremos que los nombres aparezcan en castellano, aquí sí podemos incluir en el script la instrucción **setlocale(LC_ALL,"spanish");** para traducirlos.

12.-) Teniendo en cuenta que, en general, una url tiene la estructura: *http://www.nombre de dominio/carpeta1/carpeta2/carpeta3/carpeta4..... etc.* Dada una url en una cadena de texto visualizar en cada línea las diferentes carpetas.