

Ejercicios PHP básicos

1. Pedir infinitos números enteros (hasta que se introduzca) el cero. Indicar finalmente cuál fue el máximo y cuál fue el mínimo.

Ejemplo:

```
Introduce n: -9
Introduce n: 1
Introduce n: 7
Introduce n: -10
Introduce n: 3
Introduce n: 0
Máximo: 7
Mínimo: -10
```

2. Dado un número “n”, dibujar líneas desde n caracteres hasta un carácter disminuyendo en un carácter cada línea que se dibuje. Repetir el patrón “+” “-” “.” en cada carácter que se imprima.

Ejemplo:

```
Introduce n: 7
+-+.-+
-.-+-+
-.-+-
+-+.-
-.-+
-.-
+

```

Ejemplo

```
Introduce n: 2
+-
.
```

3. Dado dos números n y p (con p entre 1 y 10), ciclar entre el cero y el p-1 tantas veces como diga n.

Ejemplo:

```
Introduce n: 3
Introduce p: 8
0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
```

4. Obtener por teclado infinitos números enteros hasta introducir el 0. Una vez hecho eso, pedir al usuario qué operación realizar, pudiendo ser ésta “sumar” o “multiplicar”, mostrándose a continuación la suma o multiplicación (respectivamente) de los datos

previamente introducidos, y en caso de que se introduzca una operación no válida, un mensaje de error.

Ejemplo:

```
Introduce n: 1
Introduce n: -7
Introduce n: 0
Operación? sumar
La suma vale -6
```

5. Similar al ejercicio 3, dado un número ciclar entre las palabras “as”, “dos”, ... “siete”, “sota”, “caballo” y “rey” tantas veces como diga tal número.

Ejemplo

```
Introduce n: 5
as dos tres cuatro cinco
```

Ejemplo:

```
Introduce n: 16
as dos tres cuatro cinco seis siete sota caballo rey as dos tres
cuatro cinco seis
```

6. Pedir por teclado nombres de personas y sus edades, hasta que se introduzca el nombre “fin”. Mostrar a continuación el nombre de cada persona introducida y su edad (entre paréntesis). NOTA: Utilizar arrays asociativos. ¿Qué ocurre si dos personas se llaman igual? ¿Cómo podríamos ver la información anterior ordenada por nombre? ¿Y por edad descendente?

Ejemplo:

```
Nombre? Ana
Edad de Ana? 18
Nombre? Pepe
Edad de Pepe? 15
Nombre? Rosa
Edad de Rosa? 18
Nombre? Juan
Edad de Juan? 17
Nombre? fin
Ana(18), Pepe(15), Rosa(18), Juan(17),
```

7. Dada una línea de texto y un número entre 1 y 6, generar una código HTML que contenga dicha línea entre tags <hX> crecientes desde 1 hasta n y luego decreciendo hasta 1 de nuevo.

Ejemplo:

```
Introduce línea de texto: HOLA
Introduce n: 4
<h1>HOLA</h1>
```

```
<h2>HOLA</h2>
<h3>HOLA</h3>
<h4>HOLA</h4>
<h3>HOLA</h3>
<h2>HOLA</h2>
<h1>HOLA</h1>
```

8. Dado un día, mes y año entre el 1 de enero de 1971 y la actualidad, mostrar el número de años, meses, días, horas, minutos y segundos transcurridos desde entonces hasta ahora (no es necesario tener en cuenta años bisiestos, y se pueden considerar todos los meses como si tuvieran 30 días)

Ejemplo:

Día: 19

Mes: 8

Año: 1980

Han transcurrido 34 años, 1 meses, 16 días, 21 horas, 25 minutos y 45 segundos desde la fecha indicada

9. Preguntar por teclado un número n (del 1 al 10), y a continuación qué formato se requiere, Si se indica "nombre", saldrán los nombres de los números del 1..N (uno, dos, tres, cuatro, ...), si se indica "romano", saldrán los números del 1..N en formato romano (i, ii, iii, iv, ...)

Diseñar el programa para que sólo tenga **un bucle** (y no tenga sentencias "if" ni "switch"), de manera que se puedan añadir nuevos formatos de números en el futuro, sin cambiar casi el código.

Ejemplo (ejecución 1):

N: 7

Formato: romano

i, ii, iii, iv, v, vi, vii

Ejemplo (ejecución 2):

N: 5

Formato: nombre

uno, dos, tres, cuatro, cinco

10. Mejorar el programa anterior para que cicle alrededor de la introducción del número N, hasta que sea un número entre 1 y 10, y para que cicle en la introducción del formato hasta que sea un formato que el programa conozca (pudiendo estar abierto a la introducción de nuevos formatos, sin necesidad de cambiar el código de comprobación)
11. Dada una cadena del estilo

"Nombre:Apellido:Telefono//OtroNombre:OtroApellido:OtroTelefono//"

Hacer un programa que muestre los datos de forma legible:

Ejemplo:

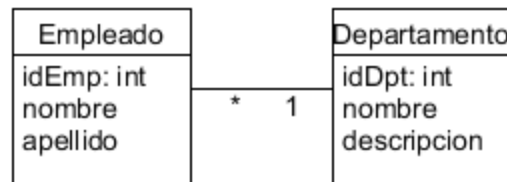
\$cadena = "Alberto:Garay:913334455//Ana:Garcia:914445566"

Nombre: Alberto
Apellido: Garay
Teléfono: 913334455

=====

Nombre: Ana
Apellido: Garcia
Teléfono: 914445566

12. Dado un texto contenido en una cadena, sustituir todas las apariciones de la palabra 'tonto' por '*****'
13. Mejorar el programa anterior para que haga esa sustitución, no sólo para la palabra 'tonto', sino para una lista de palabras prohibidas proporcionada en un array de string.
14. Utilizar un array asociativo para emular un sistema gestor de base de datos relacional, en el que poder crear una estructura como la que propone el siguiente esquema:



Crear un programa, que mediante un menú, implemente un CRUD para todos los objetos descritos. El Caso de Uso (R) que tenga dos facetas: "mostrar todos los objetos" y "Recuperar un objeto por Id.". En el caso de "mostrar todos los empleados", que en lugar del idDpt, se muestre el nombre del departamento.

SUGERENCIA: Utilizar la siguiente estructura de arrays asociativos.

