

UD1. Ejercicios de lógica

NOTA: doy por supuesto que sabéis programar, que no hace falta decir que se validan los datos y que organizáis, estructuráis, comentáis las aplicaciones.

1-Crea un programa en el que crees 4 variables: 2 cadenas y 2 números, con los siguientes valores: tu nombre, tu apellido, tu edad y tu año de nacimiento.

- Muestra en un alert las variables introducidas con formato clave valor en donde los valores cadena aparezcan envueltos entre comillas dobles y los valores numéricos entre comillas simples.
- Muestra en un alert tu nombre y apellidos separados por un salto de línea.
- Muestra en un alert la suma de las variables edad y año de nacimiento.
- Muestra en un alert la suma de todas las variables.

2-Realiza un programa que solicite números al usuario hasta que introduzca un 0. Y que muestre los valores: máximo, mínimo, suma, media y total de números introducidos.

3-Crea una función reciba un número y que dibuje un rectángulo hueco de lado del tamaño del número indicado. El valor devuelto será un array con cada una de las cadenas que forman el rectángulo.

Añade el código auxiliar necesario para probar la aplicación.

4- Crea una función reciba un número y que dibuje un triángulo de altura el tamaño del número indicado. El valor devuelto será un array con cada una de las cadenas que forman el rectángulo.

Añade el código auxiliar necesario para probar la aplicación.

5- Crea una función reciba un número y que dibuje un rombo de diagonal del tamaño del número indicado. El valor devuelto será un array con cada una de las cadenas que forman el rombo.

Añade el código auxiliar necesario para probar la aplicación.

6-Crea una función que dibuje en un alert la figura geométrica indicada por parámetro. Donde el primer parámetro será la función que genera el polígono y el segundo parámetro el número que indica el tamaño del polígono.

Deberás definir un menú que solicite el polígono a representar: cuadrado hueco, triangulo, rombo. Y después solicitar el tamaño al usuario. Incluye una opción de terminar.

Añade el código auxiliar necesario para probar la aplicación.

7-Crea una función que reciba un número e imprima la tabla de multiplicar.

Crea una función que solicite dos números entre 0 y 10, y que imprima las tablas de multiplicar entre los números indicados. Las tablas aparecerán desde el número más pequeño al mayor (aquí hay validaciones).

Añade el código auxiliar necesario para probar la aplicación.

8-Crea una función que reciba un número y calcule su factorial, la función devolverá una cadena con el desarrollo del factorial. Por ejemplo, para una entrada de 4 devolverá la cadena "4x3x2x1=24".

Añade el código auxiliar necesario para probar la aplicación. El script solicitará números al usuario hasta que no desee continuar.

9-Crea una función que reciba un número variable de parámetros y que los agrupe por tipo. Después por cada tipo se mostrará el tipo y la colección de parámetros de ese tipo incluyendo la posición original entre los parámetros.

Añade el código auxiliar necesario para probar la aplicación.

10-Crea una función que reciba una cadena de texto y que cuente el número de apariciones de cada carácter. Debe devolver un objeto de tipo clave-valor con el resultado.

Añade el código auxiliar necesario para probar la aplicación.

11- Crea una función que reciba un rango, para cada valor en el rango se informará de si el número es múltiplo de 3, si es múltiplo de 5 y si es número primo. Sólo mostrar información de los números que cumplan alguna de las condiciones.

Añade el código auxiliar necesario para probar la aplicación.

12-Juego acertar número.

Calcula un numero aleatorio entre 1 y 100, este número permanecerá secreto. Pide al usuario un numero en rango, el programa le dirá si es mayor o menor que el número secreto. Si el usuario acierta el número secreto gana. El usuario dispondrá de 5 intentos para acertar el número secreto, tras lo cual, el programa dirá que ha perdido y le mostrará el número secreto.

Añade el código auxiliar necesario para probar la aplicación.

13-Juego mates

Crea un programa que genere dos números aleatorios entre 1 y 10, y un operador aritmético. Se ofrecerá al usuario los valores para que calcule el resultado, y el programa le dirá si el resultado es correcto o incorrecto.

Las preguntas, respuesta del usuario y si ha acertado o fallado se almacenarán ordenadamente en un array.

Se preguntas se realizarán en lotes de 4, tras lo cual se consultará al usuario si desea continuar.

Cuando el usuario no desee seguir jugando se mostrará un resumen de las preguntas (número, pregunta y acierto) así como el total de preguntas acertadas y el total de falladas.

Añade el código auxiliar necesario para probar la aplicación.

14-Crea un programa de gestión académica.

Teniendo en cuenta:

- El objeto JSON de tipo alumno tiene las propiedades “nombre” y “asignaturas”.
- El objeto JSON de tipo asignatura tiene las propiedades “modulo” y “nota”.
- Disponemos módulos de “DWEK”, “DIW”, “DWES” y “DAW”.

Partiendo de un array de al menos 3 alumnos debidamente inicializados.

Crea una función que reciba el array de alumnos y que para cada alumno añada las siguientes propiedades:

- promociona, valor true si todos los módulos están aprobados, falso en caso contrario.
- media, valor medio del expediente del alumno.

Crea una función que reciba el array de alumnos modificado e imprima el listado de alumnos que promocionan con el formato “Índice-Nombre-Media”.

Crea una función que reciba el array de alumnos modificados e imprima el listado de alumnos que no promocionan con el formato “Índice-Nombre-Pendientes:[pendiente1, pendiente2, etc..]”.

Añade el código auxiliar necesario para probar la aplicación.

15-Crea un programa de gestión de personal.

La entidad “trabajador” tiene las propiedades: “código”, “nombre”, “categoría”, “contratación”.

Donde:

- Propiedad “código” tiene el formato “E01”, “E02” y es un valor único identificativo del trabajador. Se asigna automáticamente al crear el trabajador y no puede modificarse.
- Propiedad “nombre” es el nombre del trabajador.
- Propiedad “categoría” puede tomar los valores, 1, 2, 3 y representa el salario base a percibir. Donde 1-1100€, 2-1400€, 3-1900€
- Propiedad “contratación” indica el año de contratación, necesario para calcular la antigüedad.
- El importe de la nomina se calcula en base a la categoría más un 4% por cada año de antigüedad.

Se pide:

Almacenar en un array los datos de mis trabajadores.

- Listar trabajadores.
- Crear trabajador.
- Borrar trabajador, solicitando el código y confirmación.
- Modificar trabajador, solicitando el código y ofreciendo como valor por defecto el valor actual.
- Calcular nóminas. Listado ordenado por categorías de nóminas con un resumen del importe total de las nóminas de cada categoría, así como el resumen final del importe total de todas las nóminas de la empresa.

Consideraciones:

- Puede/debes tener datos cacheados.
- Implementa un menú para interactuar con la aplicación. Incluye la opción de terminar.
- Añade validaciones.
- Estructura y comenta el código.

16-Envuelve el ejercicio anterior en una función autoinvocada.