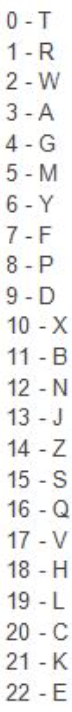
Actividades de repaso

1. Crea un fichero JavaScript e incluye las instrucciones necesarias que te permitan ejecutar la siguiente lógica:
   * Pídele su nombre y edad al usuario
   * Utiliza la estructura de control if para determinar en qué franja de edad se encuentra, de manera que consideres lo siguiente:
     1. Niño: hasta 12 años
     2. Adolescente: de 13 a 17 años
     3. Trabajador: de 18 a 64 años
     4. Jubilado: de 65 años o más
   * El programa terminará mostrando el mensaje “Juan tiene 21 años y por tanto es trabajador”.
2. Crea un fichero JavaScript e incluye las instrucciones necesarias que te permiten ejecutar la siguiente lógica de gestión de horarios de apertura:
   * Pídele al usuario la inicial del día de la semana (L, M, X, J, V, S, D)
   * Utiliza la estructura de control que consideres para mostrar el horario de apertura del día elegido. Por supuesto el horario debe ser diferente para cada día.
3. Crea un programa en JavaScript para calcular la letra de un DNI que debe introducir el usuario por teclado.



1. Escribe un programa que le pida al usuario un número, y muestre en pantalla su factorial.
2. Escribe un conversor de tiempo. El programa debe pedirle al usuario una cantidad de días, horas y minutos, y el programa devolverá la cantidad de segundos que son.
3. Crea un programa que muestre los números impares que no sean múltiplo de 3 ni de 7 que hay del 100 al 1. Realiza versiones del programa usando bucles while, do while y for.
4. Escribe un programa que ponga en práctica la siguiente lógica:
   * Mostrará el siguiente menú y el usuario deberá elegir una de las opciones:
     1. Área del triángulo (b\*h/2)
     2. Área del rectángulo (b\*h)
     3. Área del círculo (π\*r2)
     4. Salir
   * En función de la opción que elija el usuario, se le pedirán los datos necesarios en cada caso.
   * El programa calculará el resultado y los mostrará por pantalla.
   * La lógica anterior se repetirá indefinidamente hasta que el usuario pulse la tecla de salir.
5. Según la forma en la que se ejecuta JavaScript se puede decir que se trata de un lenguaje de programación:
   * 1. Compilado
     2. **Interpretado**
     3. Distribuido
     4. Combinado
6. ¿Qué utilizarías para incorporar un fichero JavaScript a una página web?
   * 1. head
     2. body
     3. **script**
     4. code
7. ¿A qué denominas prompt?
   * 1. **Al símbolo de la consola desde el que puedes lanzar instrucciones para que las ejecute el navegador**
     2. A la ruta desde la que se pueden incorporar ficheros JavaScript a un HTML
     3. A un atributo de la etiqueta que te permite incluir ficheros JavaScript externos
     4. A los mensajes de salida de la consola de un navegador
8. Selecciona la afirmación falsa en relación con las reglas que debes tener en cuenta para definir identificadores:
   * 1. **Puedes usar cualquier carácter recogido UNICODE**
     2. No distingue entre mayúsculas y minúsculas
     3. El primer carácter debe ser una letra, \_ o $
     4. No puede coincidir con una palabra reservada del lenguaje
9. Si necesitas crear una variable que no puede cambiar a lo largo de todo el programa, ¿qué tipo de variable sería la más adecuada?
   * 1. let
     2. var
     3. **const**
     4. Si no puede cambiar a lo largo del programa, no es una variable. Su propio nombre lo indica
10. ¿Qué operador usarías para calcular la potencia de un número?
    * 1. \*
      2. **\*\***
      3. \*\*\*
      4. \*\*\*\*
11. Señala la respuesta correcta en relación con la estructura de control if:
    * 1. **Puedes quitar las llaves {} si contienen una sola instrucción**
      2. No admite el encadenamiento de más de 3 if elseif else
      3. Es una estructura de control de salto
      4. Forma bucles que sin tomar precauciones podrían convertirse en infinitos
12. ¿Qué es lo más importante al diseñar un bucle del tipo que sea?
    * 1. **Asegurarse de que se puede alcanzar la condición de terminación**
      2. No abusar de su tamaño puesto que tiene costes computacionales
      3. No anidarlos para no aumentar su complejidad
      4. No olvidar colocar un default
13. ¿Qué diferencia hay entre while y do while?
    * 1. Ninguna, hacen exactamente lo mismo en todos los casos
      2. While se ejecuta siempre al menos una vez, mientras que do while no tiene por qué
      3. **Do while se ejecuta siempre al menos una vez, while no**
      4. While es una estructura repetitiva y do while de salto
14. Señala la afirmación correcta en relación con las instrucciones de salto
    * 1. Es muy buena idea usarlas con frecuencia por su agilidad saltando de un punto a otro del programa
      2. **Son muy útiles en situaciones muy concretas pero no debes abusar de ellas**
      3. A pesar de ser de salto, no permiten saltar a otros puntos del programa
      4. Son útiles, pero su complejidad hace que hayan caído en desuso