

## DWEC 2º DAW

### UNIDAD 2: MANEJO DE SINTAXIS. USO DE DATOS PRIMITIVOS

#### PRÁCTICA ESTRUCTURAS CONTROL CÓDIGO

- Realiza un programa que, según el salario actual, el número de hijos y la edad del usuario, recalcule su nuevo salario. Pedirás el nombre, salario actual (puede ser con decimales), número de hijos y la edad. Las condiciones son:

- Si gana menos de 1000€ y tiene **menos de 30 años** y tiene descendencia, su salario pasará a 1200€. En caso no tener hijos, se le aplica una subida del 5%.
- Si tiene **entre 30 y 45**, su salario se modificará de esta manera:
  - Si gana menos de 1250 y tiene más de 1 hijo → subida del 10%
  - Si gana menos de 1250 y es familia numerosa → subida del 15%
- Si tiene **más de 45 años**, independientemente de la descendencia, su salario se verá incrementado en un 15%, a excepción de sueldos iguales o mayor a 2000€.
- No se contemplan más opciones para el recálculo del salario.

Se mostrará al usuario un mensaje con su nombre, sueldo actual y nuevo sueldo.

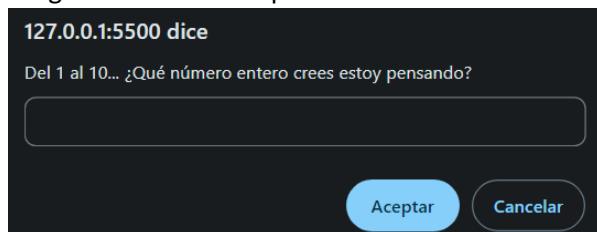
\*\*\*NO se podrán utilizar sentencias de selección anidadas

- Crea un programa que dibuje en el propio HTML una tabla. Se proporciona el fichero HTML. Debes tener en cuenta:
  - Pedirás al usuario el número de filas y columnas
  - Para dibujar filas hay que usar “`<tr>`” y “`</tr>`”
  - Para dibujar columnas hay que usar “`<td>`” y “`</td>`”
  - En cada celda de la tabla deberá poner vuestro nombre (sin apellidos)
- Realiza una función con dos parámetros. El primero será un número y el segundo el número de veces que se hallará el doble del primer parámetro. Tendremos en cuenta:
  - El número de veces que se imprimirá el doble del número por defecto será 3.
  - Antes de mostrar los números, deberemos mostrar al usuario el número que se ha pasado como argumento y el número de veces que se va a ir calculando el doble.
  - Los datos los pediremos al usuario.

```
El número es 3 y se hallará el doble 3 veces
6
12
24
```

4. Crea un programa que consista en adivinar un número aleatorio. El juego se basa en estas condiciones:

- La página calculará el número entero del 0 al 10 (`Math.random()*11`)
- Preguntará al usuario por el número.

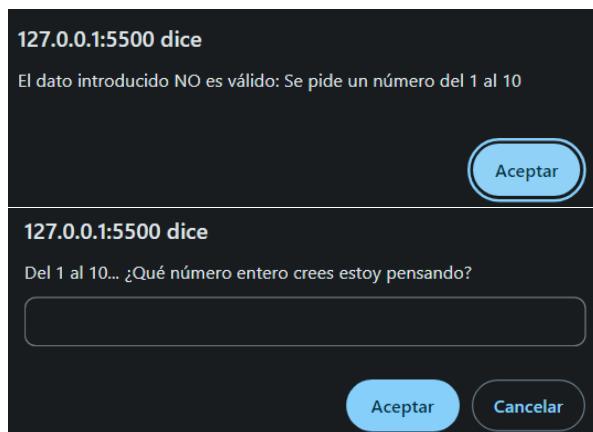


- Trabajaremos siempre con enteros, aunque el usuario ponga un número con decimales (deberás parsear el número con `parseInt()`).
- El programa llevará un contador de intentos.
- El programa verificará que el usuario:
  - No ha dejado la casilla en blanco o ha cancelado: En este caso, le preguntará si quiere seguir jugando.

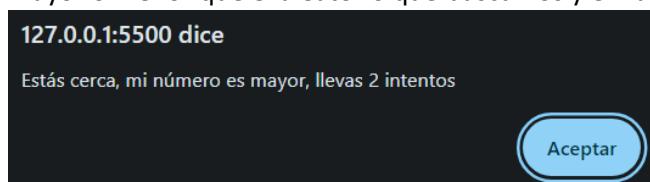


Estos intentos NO restan intentos

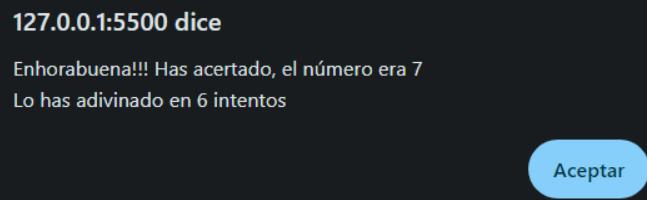
- Además verificará que es número (para ello puedes usar la función `isNaN()` que devuelve un booleano), y que está entre el rango. En cualquiera de estos casos, aparecerá el mensaje que corresponda y, volverá a darle la oportunidad directamente (mensaje inicial) de que adivine el número. Estos intentos NO restan.



- Si el usuario ha puesto un número y está en el rango, el programa le dirá si su número es mayor o menor que el aleatorio que buscamos y el número de intentos que lleva.



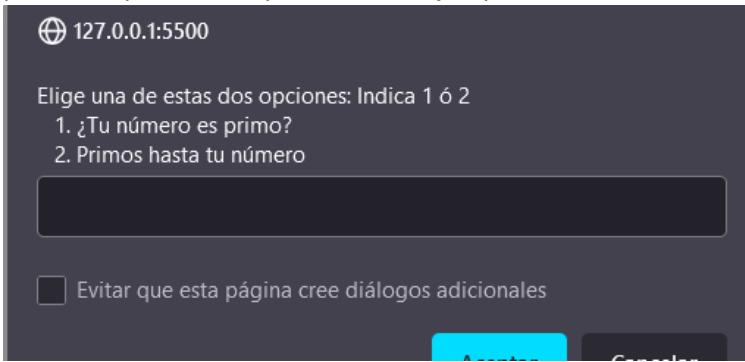
- Cuando acierta, aparece un mensaje de enhorabuena y muestra el número de intentos.



- Le preguntará al usuario si quiere volver a jugar mediante confirmación.

5. Realiza un programa que, tras interactuar con el usuario, le diga si un número es primo o no o cuántos números hay entre el 1 y el número que el usuario nos dice. Tendrás en cuenta:

- Para saber si el número es primo o no lo haremos mediante llamada a una función que has desarrollado.
- Partiremos de un “menú” en el que se dan dos opciones al usuario. Opción 1: Dame un número y te digo si es primo o no y opción 2: Dame un número y te digo qué números primos hay entre el 1 y tu número. Ejemplo:



- Si estamos en la opción 1, le diremos a través de una ventana emergente si el número X es primo o no.
- Si estamos en la opción 2, mostraremos por consola del navegador la lista de números primos. Además, deberemos indicarle cuántos son. Ejemplo:

