**EJERCICIO – 1 [ 4 puntos ]**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Realiza un programa en JavaScript que genere un número aleatorio entre un rango que se le solicitará al usuario, y permita al usuario adivinar el número en un número determinado de intentos (que también debe pedirse al usuario).

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente con confianza media

Figura .- Insertar valor mínimo

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente con confianza media

Figura .- Insertar valor máximo

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

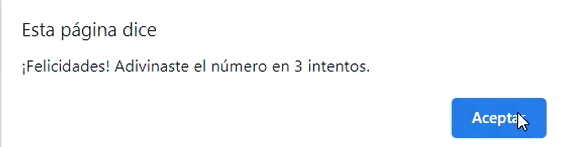
Figura .- Introducir número de intentos

El programa debe indicar si el número ingresado es mayor o menor que el número generado, además de los intentos que te quedarían:

Texto, Carta

Descripción generada automáticamente

Al final, debe mostrar el número de intentos utilizados y si el usuario adivinó el número o no.



Si no es así, se mostrará el número secreto:

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

|  |
| --- |
| **Claqueta con relleno sólido**  **[Mirad los videos proporcionados]** |

**NOTA:** Cómo se genera el número aleatorio:

var numeroAleatorio = Math.floor(Math.random() \* (max - min + 1)) + min;

**EJERCICIO – 2 [ 2.5 puntos ]**

Se proporciona un programa con algunos errores lógicos y sintácticos. Además de eso, se debe validar la entrada de usuario, es decir, se debe comprobar que realmente que el usuario está introduciendo un número entero por teclado.

**Se proporciona el código con errores lógicos y sintácticos en el aula virtual.**

**NOTA:** La validación de entrada del usuario que se pide es:

* Comprobar si el usuario ha introducido letras en lugar de números

|  |
| --- |
| Claqueta con relleno sólido  **[Mirad los videos proporcionados]** |

**EJERCICIO – 3 [ 3.5 puntos ]**

Se proporciona el código estático de una web. Sólo tenemos la estructura de la página (HTML) y los estilos CSS.

Nos piden que añadamos código dinámico (Javascript) para dotar de funcionalidades extra a la web. A continuación, se define lo que deberéis implementar:

1. El menú principal debe cambiar de color al pasar el ratón por encima de todos sus elementos.

**ATENCIÓN:** Cada elemento deberá tener un color diferente.

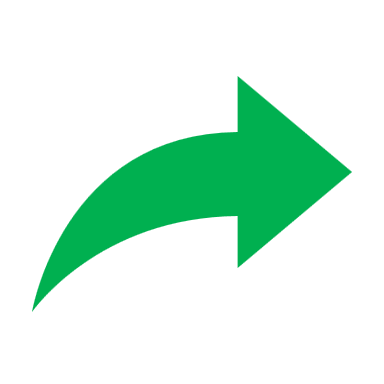


**NOTA:** Podéis utilizar colores fijos o la siguiente función, que lo que hace es obtener un color aleatorio (en hexadecimal):

|  |
| --- |
| const randomColor = Math.floor(Math.random()\*16777215).toString(16);  link.style.color = "#" + randomColor; |

1. Cambiar los colores de **TODOS** los botones (de color rosa) de la web al pasar el ratón por encima.

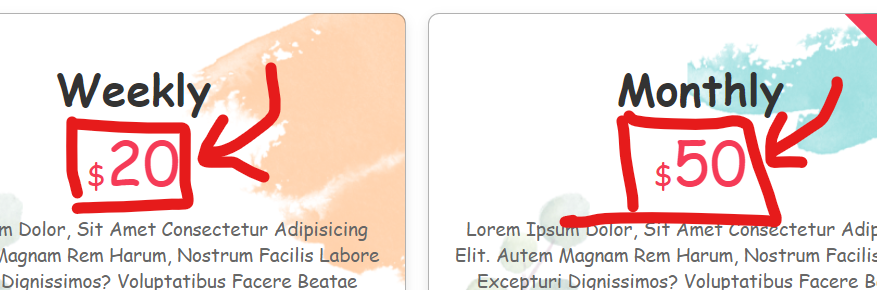
Cambiaremos a rojo (“red”).

****Texto

Descripción generada automáticamente

**OBLIGATORIO:** Utilizar el método de selección “**QuerySelectorAll()”**

1. Cambiar el tamaño de los precios al pasar el ratón por encima, en el apartado “Our pricing plans” de la web.



**NOTA\_1:** Usaremos un efecto de transición para que el efecto se vea más bonito y profesional.

Ejemplo:

price.style.transition = 'font-size 0.3s ease';

**NOTA\_2:** El tamaño actual de la fuente de estos precios es de 5.5 rem, por lo que aumentaremos dicho tamaño a 7rem.

1. En el formulario si el usuario no rellena los campos, estos cambiarán y se marcarán sus bordes en rojo:

Texto

Descripción generada automáticamente

Así:

Imagen que contiene Carta

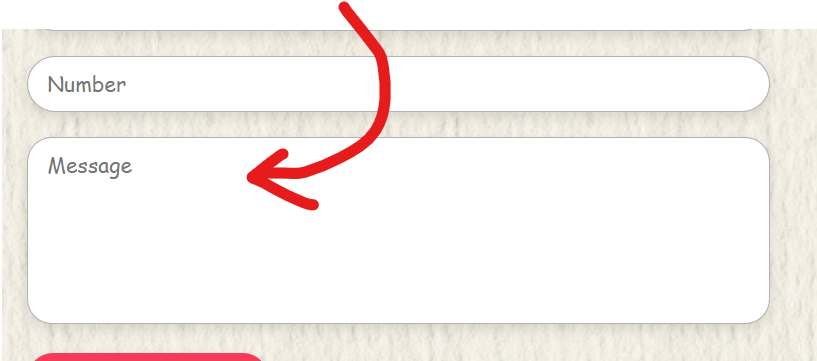
Descripción generada automáticamente

Cuando se rellene dicho input, volverá a su aspecto original:

Patrón de fondo

Descripción generada automáticamente

1. Si en el “textarea” introducimos la palabra “examendam”, el borde del campo cambiará a color verde. Si borramos la palabra, tendrá el estilo original. Recuerda que debe comportarse como se especificó en el apartado anterior (apartado 4).



Ejemplo:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

|  |
| --- |
| Claqueta con relleno sólido  **[Mirad los videos proporcionados]** |

---------------------------------------------------------------------------------------------------

****

**NOTA PARA TODO EL EJERCICIO:** NO se puede tocar el CSS.

|  |
| --- |
| **Lista de comprobación con relleno sólido**  **Puntuaciones:**   * Ejercicio 1 **(4 puntos)** * Ejercicio 2 **(2.5 puntos)** * Ejercicio 3 **(3.5 puntos)** |