



# DOCUMENTACION GENERAL

De: Pablo Andrés Calito del Cid

Indice:

## Contenido

Ejercicio 1: .....	3
Ejercicio 2: .....	4
Ejercicio 3: .....	5
Ejercicio 4: .....	6
Ejercicio 5: .....	7
Ejercicio 6: .....	8

## Ejercicio 1:

### Instrucciones:

Crea 5 funciones que impriman en consola un mensaje harcodeado, pero diferente para cada función. Asignar esas 5 funciones a un array de funciones. Hacer una función principal que ejecute aleatoriamente una función del arreglo de funciones.

### Planteamiento:

Primero que nada, debemos saber cómo hacer el código partiendo de que la mayoría del código debe ser en Java Script, podríamos utilizar distintas funciones con un mensaje predeterminado, una constante que encierre todos los mensajes, y una función que nos permita hacer una elección aleatoria y mostrarlo en la consola.

### Pasos y descripción del código implementado:

- Paso 1: Crear los componentes HTML en este caso lo conforman un div contenedor, y un div hijo, en el div contenedor colocamos las propiedades CSS para agregarle forma y centrarlo y en el div hijo agregamos todo lo que son los estilos, los otros componentes son unas 2 etiquetas <h4> y un botón para poder cambiar los mensajes, aunque también funciona recargando la página.
- Paso 2: en el paso 2 empezamos desarrollando el código en JavaScript, como lo antes mencionado creamos 5 funciones y con un `console.log ("" )` le mandamos un mensaje diferente a cada una.
- Paso 3: Creamos la función principal que se encargara de recibir los mensajes ya antes puestos en una constante, la función se encargara de escoger un mensaje aleatoriamente y mostrarlo en la consola, luego desde el botón con un `onClick` llamamos a la función.
- Paso 4: añadimos estilos desde un documento css los estilos son personalizables al gusto en mi caso utilice una maqueta de colores rojos, grises y blancos, añadiéndole con un `:hover` animaciones a los botones.

## Ejercicio 2:

### Instrucciones:

Crea una función que determine si una cadena de texto es un palíndromo, es decir, si se lee igual de izquierda a derecha y de derecha a izquierda. La función debe devolver true si es un palíndromo y false en caso contrario.

### Planteamiento:

Primero debemos saber que un palíndromo es una palabra que se lee igual de izquierda a derecha que de derecha a izquierda, ahora que sabemos deberíamos definir 2 funciones una para verificar que sea un palíndromo y otra para validar posibles equivocaciones como por ejemplo que no se haya ingresado nada o que sea o no sea un palíndromo y poder mostrarlo en un text área.

### Pasos y descripción del código implementado:

- Paso 1: Crear los componentes HTML en este caso lo conforman un div contenedor, y un div hijo, en el div contenedor colocamos las propiedades CSS para agregarle forma y centrarlo y en el div hijo agregamos todo lo que son los estilos, los otros componentes son unas 2 etiquetas <h1> para el título, también usamos un input para escribir la palabra, y utilizamos un botón para verificar la palabra, usamos un text área para mostrar las respuesta con un onClick para llamar a la función de JavaScript.
- Paso 2: en el paso 2 empezamos desarrollando el código en JavaScript, como lo antes mencionado creamos 2 funciones una para que al momento de verificar sepa que es un palíndromo y la otra para validar los mensajes.
- Paso 3: usamos un Split.reverse.join para que pueda leer de ambos lados de la palabra junto con un toLowerCase.replace, luego la función para validar si es o no es un palíndromo
- Paso 4: añadimos estilos desde un documento css los estilos son personalizables al gusto en mi caso utilice una maqueta de colores rojos, grises y blancos, añadiéndole con un :hover animaciones a los botones.

## Ejercicio 3:

### Instrucciones:

Crea una función que cuente el número de vocales y consonantes en una cadena de texto. Luego, muestra los resultados en la consola.

### Planteamiento:

Partimos creando nuestros archivos, HTML, CSS, JS, luego empezamos creando funciones, para obtener el valor del campo, definimos las variables, y también definimos las vocales, creamos un for y un if para crear la función que cuente las vocales y consonantes.

### Pasos y descripción del código implementado:

- Paso 1: Crear los componentes HTML en este caso lo conforman un div contenedor, y un div hijo, en el div contenedor colocamos las propiedades CSS para agregarle forma y centrarlo y en el div hijo agregamos todo lo que son los estilos, los otros componentes son un input para escribir la palabra, un botón para aceptar la palabra y dos etiquetas <p> donde se mostrara el conteo
- Paso 2: en el paso 2 empezamos desarrollando el código en JavaScript, como lo antes mencionado creamos una función que encerrara todo, luego procedemos a crear una constante que obtenga el valor del campo y lo convierta a letras minúsculas, luego declaramos las variables y en una le asignamos como valor las vocales.
- Paso 3: Luego con un For recorreremos cada carácter de la palabra, verificamos con un if si es letra minúscula o mayúscula, si el carácter esta incluido en las vocales incrementara e valor del contador de vocales al igual si esta incluido en las consonantes incrementara el valor del contador de consonantes.
- Paso 4: añadimos estilos desde un documento css los estilos son personalizables al gusto en mi caso utilice una maqueta de colores rojos, grises y blancos, añadiéndole con un :hover animaciones a los botones.

## Ejercicio 4:

### Instrucciones:

Validación de Contraseña: Crea una función que valide si una contraseña cumple con ciertos criterios, como tener al menos 8 caracteres, contener letras mayúsculas, letras minúsculas y al menos un número.

### Planteamiento:

Este código verifica la seguridad de una contraseña en términos de longitud, letras mayúsculas, letras minúsculas y números, mostrando un mensaje de error en rojo si no cumple con los criterios o un mensaje de "Contraseña correcta" en verde si es válida.

### Pasos y descripción del código implementado:

- Paso 1: Crear los componentes HTML en este caso lo conforman un div contenedor, y un div hijo, en el div contenedor colocamos las propiedades CSS para agregarle forma y centrarlo y en el div hijo agregamos todo lo que son los estilos, los otros componentes son unas 2 etiquetas <h1>, 2 label, un input, un botón.
- Paso 2: en el paso 2 empezamos desarrollando el código en JavaScript, como lo antes mencionado obtenemos la contraseña ingresada, y verificamos la longitud máxima, verificamos con if que las letras mayúsculas y minúsculas y números estén bien
- Paso 3: si todo es correcto un un mensale en el label color verde mostrara como correcto de lo contrario mostrara un mensaje color rojo mostrando incorrecto.
- Paso 4: añadimos estilos desde un documento css los estilos son personalizables al gusto en mi caso utilice una maqueta de colores rojos, grises y blancos, añadiéndole con un :hover animaciones a los botones.

## Ejercicio 5:

### Instrucciones:

Crear una función que devuelva un número entero que corresponda a la cantidad de dígitos en el número entero proporcionad.

### Planteamiento:

Este código cuenta la cantidad de dígitos en un número ingresado y muestra el resultado en un área de texto de la página web en cuatro pasos:

1. Obtener el número ingresado.
2. Verificar si el número es válido.
3. Contar los dígitos convirtiendo el número en una cadena de texto y tomando su longitud.
4. Mostrar el resultado o un mensaje de error en el área de texto.

### Pasos y descripción del código implementado:

- Paso 1: Crear los componentes HTML en este caso lo conforman un div contenedor, y un div hijo, en el div contenedor colocamos las propiedades CSS para agregarle forma y centrarlo y en el div hijo agregamos todo lo que son los estilos, los otros componentes son unas 2 etiquetas <h1>, un input para ingresar el número, un botón para contar los dígitos, y un TextArea para mostrar el resultado del conteo.
- Paso 2: Obtenemos el numero ingresado con el id también se convierte en un entero con un parseInt y se almacena en una variable, verifica si el numero es valido con un else if y hace la validación.
- Paso 3: cuenta los numero con la función contarDigitosNumero luego del conteo convierte el numero a texto otra vez y los muestra en el textarea de HTML.
- Paso 4: añadimos estilos desde un documento css los estilos son personalizables al gusto en mi caso utilice una maqueta de colores rojos, grises y blancos, añadiéndole con un :hover animaciones a los botones.

## Ejercicio 6:

### Instrucciones:

Crea una función que tome dos fechas y devuelva el número de días entre la primera y la segunda fecha.

### Planteamiento:

Este código toma dos fechas ingresadas por el usuario, calcula la diferencia en días entre ellas y muestra el resultado en un área de texto en la página web, o muestra un mensaje de error si las fechas no son válidas.

### Pasos y descripción del código implementado:

- Paso 1: Crear los componentes HTML en este caso lo conforman un div contenedor, y un div hijo, en el div contenedor colocamos las propiedades CSS para agregarle forma y centrarlo y en el div hijo agregamos todo lo que son los estilos, los otros componentes son unas 2 etiquetas `<h1>` para los títulos, junto con dos labels, dos inputs, un botón, y un text área para mostrar el resultado.
- Paso 2: en el paso 2 empezamos desarrollando el código en JavaScript, primero obtenemos los elementos relevantes de HTML mediante el id, partiendo de eso convertimos las fechas ingresadas a objetos fecha, y verificamos que sean validas mediante un else if, si alguna fecha no es válida marcara error.
- Paso 3: Calculamos la diferencia entre fechas con una resta, y luego convertimos la diferencia a días esto teniendo el resultado de la resta de las fechas en una variable luego a la variable lo dividimos entre  $(1000*60*60*24)$  esto da el resultado en días y mostramos el resultado en el text área.
- Paso 4: añadimos estilos desde un documento css los estilos son personalizables al gusto en mi caso utilice una maqueta de colores rojos, grises y blancos, añadiéndole con un `:hover` animaciones a los botones.