Memoria Hito Individual 3er Trimestre.

Trabajo realizado por: Álvaro Serrano Íñigo

Ejercicio 1 HTML:

El archivo es un fragmento de código HTML que contiene una calculadora básica con botones para realizar operaciones matemáticas. Está contenido en una etiqueta div con la clase "container".

Dentro de este div, hay un elemento p que contiene el texto "Empieza a calcular". A continuación, hay otro div con la clase "calculator", que contiene los elementos de la calculadora.

El primer elemento dentro de este div es un div con la clase "calculator\_\_display", que muestra el resultado de los cálculos. El valor inicial de este elemento es 0.

A continuación, hay otro div con la clase "calculator\_\_keys", que contiene todos los botones de la calculadora. Hay botones para los números del 0 al 9, botones para los operadores de suma, resta, multiplicación y división, un botón para la coma decimal, un botón para borrar la pantalla, un botón para calcular el resultado, y botones para las operaciones de porcentaje, raíz cuadrada y potenciación.

El archivo también incluye un script llamado "script.js" que se encuentra en la misma carpeta y que se utiliza para controlar el comportamiento de la calculadora.

Ejercicio 1 CSS:

Este es un archivo CSS que contiene una serie de reglas de estilo para la apariencia y el diseño de un sitio web. A continuación se detallan las reglas y su función:

El selector universal (\*) y los pseudo-elementos ::before y ::after se establecen en box-sizing: inherit, lo que hereda la propiedad box-sizing del elemento padre. El selector html se establece en box-sizing: border-box, lo que hace que el tamaño total de los elementos incluya el relleno y el borde.

El elemento body se establece en margin: 0, lo que elimina cualquier margen predeterminado.

Los elementos embed, iframe, img, object y video se establecen en max-width: 100%, lo que asegura que se ajusten al ancho del contenedor padre y sean responsivos.

Los encabezados (h1-h6), las listas (ul y ol), los párrafos (p), el código preformateado (pre), las citas (blockquote), las figuras y las líneas horizontales (hr) se establecen en margin: 0 y padding-right: 0 y padding-left: 0, lo que elimina cualquier margen y relleno predeterminado.

Los enlaces (a) se establecen en text-decoration: none, lo que elimina la subrayado por defecto de los enlaces. También se establece a:focus en outline: none, lo que elimina el contorno del elemento cuando está enfocado.

Los encabezados (h1-h6) se establecen en display: block, lo que asegura que los encabezados ocupen una línea completa y no se superpongan con otros elementos.

Las listas ordenadas y no ordenadas (ol y ul) se establecen en list-style: none, lo que elimina los puntos predeterminados y cualquier marcador de lista.

Los elementos de formulario como los campos de entrada (input), las áreas de texto (textarea) y los botones se establecen en border: 0, border-radius: 0, background-color: transparent, font-size: inherit, font-family: inherit, font-weight: inherit y outline: none. Esto elimina cualquier borde y redondez predeterminado y hereda la fuente y el tamaño de fuente de su elemento padre. También se establece appearance: none, lo que elimina la apariencia predeterminada del elemento en algunos navegadores, y text-align: left para alinear el texto del elemento a la izquierda.

Los elementos de formulario y los botones se establecen en input:hover, input:active, input:focus, textarea:hover, textarea:active, textarea:focus, button:hover, button:active y button:focus en outline: none, lo que elimina cualquier contorno adicional en los elementos cuando se seleccionan o se interactúa con ellos.

:root establece la fuente predeterminada para la página en Helvetica, Arial, sans-serif.

html se establece en font-size: 175%, font-weight: 300 y line-height: 1.3. Esto aumenta el tamaño de la fuente en un 175% y establece la fuente en un peso ligero.

body se establece en align-items: center, display: flex, justify-content: center y height: 100vh. Esto centra cualquier contenido en el centro de la pantalla y establece la altura del cuerpo en el 100% del viewport.

.container se establece en max-width: 20em y .container > p se establece en text-align: center,

lo que limita el ancho máximo del contenedor a 20em y centra el texto dentro del contenedor.

.calculator se establece en border-radius: 12px y box-shadow: 0 0 40px 0px rgba(0, 0, 0, 0.15). Esto le da al elemento de la calculadora bordes redondeados y una sombra suave.

.calculator\_\_display se establece en background-color: #222222, color: #fff, font-size: 1.714285714em y padding: 0.5em 0.75em. Esto establece el estilo del área de visualización de la calculadora con un fondo oscuro, texto blanco y un tamaño de fuente específico.

.calculator\_\_keys se establece en background-color: #999, display: grid, grid-gap: 1px y grid-template-columns: repeat(4, 1fr). Esto establece el estilo de las teclas de la calculadora con un fondo gris, utiliza una cuadrícula para posicionar las teclas y divide las columnas en 4 partes iguales.

.calculator\_\_keys > \* se establece en background-color: #fff, padding: 0.5em 1.25em, position: relative y text-align: center. Esto establece el estilo de cada tecla de la calculadora con un fondo blanco, un relleno específico y alinea el texto en el centro.

.calculator\_\_keys > \*:active::before y .calculator\_\_keys > .is-depressed::before se establecen en varios estilos para crear un efecto visual cuando se presiona una tecla. Esto incluye un fondo semitransparente y una sombra.

.key--operator se establece en background-color: #eee. Esto establece el estilo de las teclas de operador de la calculadora con un fondo gris claro.

.key--equal se establece en background-image: linear-gradient(to bottom, #fe886a, #ff7033), grid-column: -2 y grid-row: 2 / span 4. Esto establece el estilo de la tecla de igual de la calculadora con un degradado de color como fondo y lo posiciona en la segunda columna desde la derecha y en las filas 2 a 4 de la cuadrícula.

Ejercicio 1 JavaScript:

Este archivo contiene código JavaScript para una calculadora simple. En primer lugar, se selecciona el elemento HTML de la calculadora y se guardan las teclas y la pantalla en variables separadas utilizando el método querySelector().

A continuación, se definen tres variables para almacenar los valores que se van a utilizar en las operaciones de la calculadora: firstValue, operator y awaitingNextValue. firstValue almacena el primer número que se ingresa en una operación, operator almacena el tipo de operación que se realizará y awaitingNextValue se establece en true cuando se espera que se ingrese el segundo valor en una operación.

Se define la función calculate() para realizar las operaciones matemáticas. La función toma tres argumentos: first, operator y second. Convierte los valores de first y second a números de punto flotante y realiza la operación correspondiente según el operador recibido.

Después, se agrega un event listener al elemento de las teclas para detectar cuando se hace clic en una tecla. Cuando se hace clic en una tecla, se obtiene el contenido de la tecla y se comprueba si se trata de una acción especial o de un número.

Si la tecla es un número, se verifica si se debe agregar al número actual en la pantalla o si se debe reemplazar por el número ingresado. Si la tecla es un punto decimal, se comprueba si el número actual ya tiene un decimal o no.

Si la tecla es una operación matemática, se verifica si se ha ingresado un primer valor y un operador anteriormente. Si es así, se calcula el resultado y se muestra en la pantalla. Si no, se guarda el número actual como primer valor y se establece el operador para la próxima operación.

Si la tecla es la tecla "C" para borrar, se restablecen los valores y se borra la pantalla. Si la tecla es la tecla "=" para calcular el resultado, se verifica si se ha ingresado un primer valor y un operador anteriormente. Si es así, se calcula el resultado y se muestra en la pantalla, y se restablecen los valores para la próxima operación.

En general, este archivo es un código JavaScript que implementa una calculadora simple con operaciones matemáticas básicas.

Ejercicio 2 HTML:

El archivo consiste en una página web que contiene una tabla de información de provincias, así como un botón para leer datos en formato JSON y una sección de pie de página que muestra la fecha y hora actualizada constantemente. La estructura de la página está organizada en dos secciones principales, la primera es un contenedor con la clase "container" y la segunda es una sección con la clase "section". Dentro de esta sección se encuentra una etiqueta "h1" que muestra el título "Provincia", seguido de un botón que tiene la clase "btn" y el id "boton", que cuando se presiona, ejecuta una función llamada "traerDatos()" para leer datos en formato JSON.

Luego de esto, hay una etiqueta "br" que se utiliza para saltar una línea y otra etiqueta "br" para hacerlo nuevamente. Después de estas etiquetas, se encuentra una tabla que contiene una cabecera con cinco columnas: "CODPROV", "NOMBRE\_PROVINCIA", "CODAUTON", "COMUNIDAD\_CIUDAD\_AUTONOMA" y "CAPITAL\_PROVINCIA". La información de las provincias se mostrará en el cuerpo de la tabla, que está vacío inicialmente y que será llenado con los datos del archivo JSON cuando se presione el botón.

Por último, en la sección de pie de página, hay un script que define una función llamada "actualizarFechaHora()" que muestra la fecha y hora actual. Esta función se ejecuta inmediatamente al cargar la página y se actualiza cada segundo con la ayuda de la función "setInterval()".

Ejercicio 2 CSS:

Este archivo contiene un conjunto de reglas CSS que se aplican al cuerpo del documento HTML, así como a diferentes elementos que se utilizan en la página. En primer lugar, se establece que tanto el cuerpo como el elemento HTML ocuparán el 100% del ancho y alto de la ventana del navegador.

A continuación, se definen algunas propiedades para el cuerpo y los elementos de párrafo. Se establece un color de texto y una fuente, que es "Open Sans" en estilo sans-serif. Luego, se definen las propiedades de estilo para los encabezados h1 y h2. El h1 se define con un color de texto diferente al resto del cuerpo, un peso de fuente más grande y un tamaño de fuente y línea que es 1 rem y 3.25 rem respectivamente, lo que significa que el texto es relativamente grande y espaciado. El h2 también tiene un color de texto diferente, una fuente más ligera y un tamaño de fuente y línea que es 0.5 rem y 2.5 rem respectivamente.

Por último, se definen algunas reglas CSS para el contenedor. Se utiliza la propiedad "box-sizing" con el valor "content-box" para establecer el modelo de caja como el modelo de contenido, y se establece el ancho del contenedor en un 100% del ancho de la ventana del navegador. Esto significa que el contenedor ocupará todo el ancho disponible en la ventana del navegador.

Ejercicio 2 JSON:

Este es un archivo JSON que contiene información sobre las provincias de España. Cada elemento del archivo representa una provincia y contiene cuatro campos: CODPROV, NOMBRE\_PROVINCIA, CODAUTON, COMUNIDAD\_CIUDAD\_AUTONOMA, y CAPITAL\_PROVINCIA.

Ejercicio 2 Javascript:

La función traerDatos() utiliza el objeto XMLHttpRequest() para realizar una solicitud GET a un archivo JSON alojado en http://localhost/HI3/HI3\_2/datos/datos.json.

La solicitud es asíncrona (true), lo que significa que la función no se bloqueará mientras se espera la respuesta del servidor.

La función define un manejador de eventos onreadystatechange que se ejecuta cada vez que cambia el estado de la solicitud. Si el estado es 4 (solicitud completada) y el código de estado es 200 (éxito), se parsea el objeto JSON devuelto con JSON.parse() y se itera sobre sus elementos para construir una tabla en HTML con los datos obtenidos.

La tabla se inserta en el elemento con el ID resp del documento HTML. Si el elemento resp ya contiene contenido, se sobrescribe.