

KRAKE DEV

EVALUACIÓN INDIVIDUAL

DESCRIPCIÓN GENERAL

Se creará una página que permita agregar elementos a un arreglo de personas y mostrarlos en pantalla en una tabla. Luego se aplicará un algoritmo para buscar a la persona de mayor edad.

CALIFICACIÓN

La calificación total es sobre 100 puntos, se requiere un mínimo de 60 puntos para aprobar la evaluación.

PARTE 1 - AGREGAR PERSONA 40 PUNTOS

1. Crear una carpeta con su nombre y apellido, dentro de la carpeta, crear los archivos orden.html, orden.js e incluir el archivo utilitarios.js.
2. En orden.js agregar un arreglo de personas, con los siguientes objetos:

0	<code>nombre:"Marcos" edad:18</code>
1	<code>nombre:"Roberto" edad:15</code>
2	<code>nombre:"Kate" edad:25</code>
3	<code>nombre:"Diana" edad:12</code>
4	<code>nombre:"Benja" edad:5</code>

3. En el html, utilizar el código de alguno de sus ejercicios que tenga estilos y permita mostrar los errores de validación en la parte inferior de las cajas de texto. Agregar los estilos.
4. Agregar dos cajas de texto: una para ingresar el nombre de la persona y otra para ingresar la edad.
5. Agregar un botón GUARDAR
6. Crear una función agregarPersona, no recibe parámetros y realiza la siguiente lógica:
 - a. Toma de la pantalla el nombre de la persona, valida que tenga al menos 3 caracteres, si no cumple la condición muestra el error en pantalla
 - b. Toma de la pantalla la edad de la persona, valida que sea un entero entre 0 y 100, si no cumple la condición muestra el error en pantalla
 - c. Solamente si pasa todas las validaciones:
 - i. Crea un objeto nuevaPersona, sin atributos.
 - ii. Agrega a nuevaPersona los atributos nombre y edad al objeto, con los valores recuperados de la pantalla
 - iii. Agrega el objeto al arreglo personas

iv. Muestra un alert PERSONA AGREGADA CORRECTAMENTE

PARTE 2 - MOSTRAR PERSONAS

30 PUNTOS

Muestra una tabla en pantalla con todas las personas del arreglo, el orden de las columnas es: EDAD NOMBRE

La tabla se debe mostrar apenas se carga la página (evento onload, en el body)

La tabla se debe refrescar cada vez que se agrega una persona

Agregar estilos a la tabla

PARTE 3 - BUSCAR MAYOR

30 PUNTOS

En esta sección se buscará el elemento de mayor edad, de acuerdo al siguiente algoritmo:

ALGORITMO PARA ENCONTRAR EL MAYOR

Se tiene el siguiente arreglo de objetos persona, y se quiere determinar el de mayor edad

0	<u>nombre:"Marcos"</u> edad:18
1	<u>nombre:"Roberto"</u> edad:15
2	<u>nombre:"Kate"</u> edad:25
3	<u>nombre:"Diana"</u> edad:12
4	<u>nombre:"Benja"</u> edad:5

Para encontrarlo se aplicará el siguiente algoritmo:

1. Se tiene una variable **personaMayor** en la cual se va a guardar la persona con mayor edad en el arreglo. El algoritmo inicia asignando a **personaMayor**, el primer elemento del arreglo, es decir el objeto en la posición **0**

personaMayor



nombre:"Marcos" edad:18

2. Se procede a comparar la **edad** de la personaMayor con cada uno de los demás objetos del arreglo.

0	<u>nombre:"Marcos"</u> edad:18	
1	<u>nombre:"Roberto"</u> edad:15	
2	<u>nombre:"Kate"</u> edad:25	
3	<u>nombre:"Diana"</u> edad:12	
4	<u>nombre:"Benja"</u> edad:5	

En cada comparación, se verifica: Si la edad del objeto del arreglo es mayor a la edad de lo que tenemos guardado en **personaMayor** en ese momento, entonces este objeto pasa a ser la nueva **personaMayor**.

Así:

Cuando se recupera la persona en la **posición 1**:

Compara: la edad de Roberto(15) es mayor a la edad de Marcos (18)?

Como no se cumple la condición, Marcos sigue siendo la personaMayor

Cuando se recupera la persona en la **posición 2**:

Compara: la edad de Kate(25) es mayor a la edad de Marcos (18)?

Como si es mayor, Kate pasa a ser la nueva la personaMayor

personaMayor



nombre:"Kate" edad:25

Cuando se recupera la persona en la **posición 3**:

Compara: la edad de Diana(12) es mayor a la edad de Kate(25)?

Como no se cumple la condición, Kate sigue siendo la personaMayor

Cuando se recupera la persona en la **posición 4**:

Compara: la edad de Benja(5) es mayor a la edad de Kate(25)?

Como no se cumple la condición, Kate sigue siendo la personaMayor

3. Al terminar de barrerse todos los elementos del arreglo, la variable personaMayor tiene la persona con la mayor edad de todo el arreglo.

IMPLEMENTACIÓN

1. Crear una función encontrarMayor, no recibe parámetros y realiza la siguiente lógica:
 - Declarar una variable **personaMayor**, asignar el primer elemento del arreglo.
 - Declarar una variable **elementoPersona**, para acceder a cada uno de los objetos del arreglo.
 - Con un for, barrer todos los elementos del arreglo, empezando en la posición 1.

- En cada iteración, recuperar la persona en la variable elemento persona, e imprimir en consola el nombre y la edad de la persona a la que está accediendo.
2. Agregar el botón ENCONTRAR MAYOR
 3. Crear una función determinarMayor, que no recibe parámetros, e invoca a la función encontrarMayor.
 4. Desde el botón invocar a determinarMayor
 5. Hasta el momento, cuando presiona el botón, solamente debe imprimir los elementos del arreglo en consola.
 6. En la función encontrarMayor realizar los siguientes cambios:
 - En cada iteración, comparar si la edad del elemento al que se está accediendo en el arreglo es mayor a la edad de personaMayor, entonces este objeto pasa a ser en nuevo mayor y por tanto se debe asignar a personaMayor
 - Luego del for, retornar personaMayor
 7. Desde determinarMayor, guardar el retorno de personaMayor en una variable llamada **mayor**. Mostrar en pantalla el nombre y la edad de la persona mayor.
 8. Con una lógica similar a encontrarMayor, crear la función encontrarMenor, el único cambio es que va a encontrar y retornar la persona con menor edad del arreglo.
 9. Con una lógica similar a determinarMayor, crear la función determinarMenor, mostrar en pantalla el nombre y edad de la persona con menor edad del arreglo
 10. Invocar determinarMenor desde un botón ENCONTRAR MENOR agregado en la pantalla.