******

***Actividad 4 - Taller Práctico de JavaScript***

***ALEXANDER GRIMALDO ESTUPIÑAN.***

ID Banner: 100097037

Corporación universitaria iberoamericana, Ingeniería de software

Diplomado Desarrollo WEB

ING. Camila Quimbaya

Agosto – 2024

***ENLACE DEL PROYECTO GITHUB***

<https://github.com/grimi40/ACTIVIDAD-4-TALLER-JAVA-.git>

1. ***Crea una función llamada esBisiesto que reciba un parámetro año.***

***La función debe devolver true si el año esbisiesto y false en caso contrario.***

***Un año esbisiesto si es divisible por 4,pero no por 100,a menos que también sea divisible por 400.***

function esBisiesto(año) {

// Un año es bisiesto si:

// - Es divisible por 400.

// - Es divisible por 4 y no es divisible por 100.

return (año % 400 === 0) || (año % 4 === 0 && año % 100 !== 0);

}

Explicación de la función:

año % 400 === 0: Esta parte verifica si el año es divisible exactamente por 400. Si lo es, definitivamente es bisiesto.

año % 4 === 0 && año % 100 !== 0: Esta parte verifica si el año es divisible por 4 pero no por 100. Si cumple esta condición, también es bisiesto.

||: El operador || (o lógico) se utiliza para combinar ambas condiciones. Si alguna de las dos se cumple, la función devuelve true.

const año = 2024;

if (esBisiesto(año)) {

console.log(año + " es un año bisiesto.");

} else {

console.log(año + " no es un año bisiesto.");

}

***2 ) Crea una función llamada celsiusAFahrenheit que reciba un parámetro***

***celsius.***

***La función debe devolver el equivalente en grados Fahrenheit.***

function celsiusAFahrenheit(celsius) {

// Fórmula para convertir Celsius a Fahrenheit: (Celsius \* 9/5) + 32

return (celsius \* 9/5) + 32;

}

Explicación:

Parámetro celsius: Esta función toma como entrada un número que representa la temperatura en grados Celsius.

Fórmula de conversión: La fórmula utilizada es la estándar para convertir Celsius a Fahrenheit: se multiplica la temperatura en Celsius por 9/5 y se le suma 32.

Retorno: La función devuelve el resultado de la conversión, que es la temperatura equivalente en grados Fahrenheit.

EJEMPLO DE USO

let temperaturaCelsius = 25;

let temperaturaFahrenheit = celsiusAFahrenheit(temperaturaCelsius);

console.log(temperaturaCelsius + " grados Celsius equivalen a " + temperaturaFahrenheit + " grados Fahrenheit.");

3). ***Crea una función llamada mayorDeDos que reciba dos parámetros n1 y n2. La función debe devolver el número mayor.***

function mayorDeDos(n1, n2) {

// Si n1 es mayor o igual que n2, retorna n1

if (n1 >= n2) {

return n1;

} else {

// Si no, retorna n2

return n2;

}

}

Explicación:

Parámetros: La función mayorDeDos toma dos números, n1 y n2, como entrada.

Comparación: Se compara si n1 es mayor o igual que n2 utilizando el operador >=.

Retorno:

Si n1 es mayor o igual, la función devuelve n1 como el número mayor.

Si n1 es menor que n2, la función devuelve n2 como el número mayor.

EJEMPLO DE USO

let numero1 = 15;

let numero2 = 20;

let numeroMayor = mayorDeDos(numero1, numero2);

console.log("El número mayor es:", numeroMayor);

4***. Crea una función llamada convertirMinutos que reciba un parámetro minutos. La función debe devolver una cadena con el formato "X horas y Y minutos".***

function convertirMinutos(minutos) {

// Calculamos las horas dividiendo los minutos entre 60 y obteniendo la parte entera

const horas = Math.floor(minutos / 60);

// Calculamos los minutos restantes restando a los minutos totales el producto de las horas por 60

const minutosRestantes = minutos % 60;

// Construimos la cadena con el formato deseado

return horas + " horas y " + minutosRestantes + " minutos";

}

Explicación:

Parámetro minutos: La función recibe como entrada un número que representa la cantidad total de minutos.

Cálculo de horas:

Se divide el número total de minutos entre 60.

La función Math.floor() se utiliza para obtener la parte entera de la división, que representa el número de horas completas.

Cálculo de minutos restantes:

Se calcula el resto de la división entre minutos y 60. Este resultado representa los minutos que sobran después de completar las horas.

Construcción de la cadena:

Se crea una cadena concatenando las horas, la palabra "horas", los minutos restantes y la palabra "minutos".

EJEMPLO DE USO

const totalMinutos = 135;

const resultado = convertirMinutos(totalMinutos);

console.log(resultado); // Imprimirá: "2 horas y 15 minutos"

5. ***Crea una función llamada estaEnRango que reciba tres parámetros:***

***numero, inicio y fin.***

***La función debe devolver true si el numero está dentro del rango [inicio,***

***fin], y false en caso contrario.***

function estaEnRango(numero, inicio, fin) {

// Verificamos si el número es mayor o igual al inicio y menor o igual al fin

return numero >= inicio && numero <= fin;

}

Explicación:

Parámetros:

numero: El número que queremos verificar.

inicio: El límite inferior del rango.

fin: El límite superior del rango.

Lógica:

La función utiliza el operador lógico && (y) para combinar dos condiciones:

numero >= inicio: Verifica si el número es mayor o igual al inicio del rango.

numero <= fin: Verifica si el número es menor o igual al fin del rango.

Si ambas condiciones se cumplen, el número está dentro del rango y la función devuelve true. De lo contrario, devuelve false.

EJEMPLO DE USO

let numero = 5;

let inicio = 2;

let fin = 7;

if (estaEnRango(numero, inicio, fin)) {

console.log("El número está dentro del rango.");

} else {

console.log("El número está fuera del rango.");

}

6. ***Crea una función llamada calcularPrecioFinal que reciba dos parámetros: precioInicial y descuento. La función debe devolver el precio final después de aplicar el descuento.***

function calcularPrecioFinal(precioInicial, descuento) {

// Convertimos el porcentaje de descuento a un decimal

const descuentoDecimal = descuento / 100;

// Calculamos el descuento a aplicar

const descuentoCantidad = precioInicial \* descuentoDecimal;

// Calculamos el precio final restando el descuento al precio inicial

const precioFinal = precioInicial - descuentoCantidad;

return precioFinal;

}

**Explicación:**

1. **Parámetros:**
   * precioInicial: El precio original del producto.
   * descuento: El porcentaje de descuento a aplicar.
2. **Conversión a decimal:**
   * Se divide el descuento entre 100 para obtener su representación decimal (por ejemplo, 20% se convierte en 0.2).
3. **Cálculo del descuento:**
   * Se multiplica el precioInicial por el descuentoDecimal para obtener la cantidad exacta del descuento.
4. **Cálculo del precio final:**
   * Se resta la descuentoCantidad del precioInicial para obtener el precio final.
5. **Retorno:**
   * La función devuelve el precioFinal calculado.

**Ejemplo de uso:**

const precioOriginal = 100;

const descuentoPorcentaje = 20;

const precioConDescuento = calcularPrecioFinal(precioOriginal, descuentoPorcentaje);

console.log("El precio final es:", precioConDescuento); // Imprimirá: "El precio final es: 80"

**7*.) Crea una función llamada puedeVotarque reciba un parámetro edad.***

***La función debe devolver true si la edad es 18 o mayor, y false en caso***

***contrario.***

function puedeVotar(edad) {

// Si la edad es mayor o igual a 18, la persona puede votar

return edad >= 18;

}

Explicación:

Parámetro edad: La función recibe como entrada un número que representa la edad de la persona.

Condición: Se verifica si la edad es mayor o igual a 18. Si se cumple esta condición, la persona puede votar.

Retorno: La función devuelve un valor booleano (true o false). Si la edad es válida para votar, se devuelve true, de lo contrario, se devuelve false.

Ejemplo de uso:

let edadPersona = 20;

if (puedeVotar(edadPersona)) {

console.log("La persona puede votar.");

} else {

console.log("La persona no puede votar.");

}

***8.) Crea una función llamada calcularAreaTriangulo que reciba dos parámetros: base y altura. La función debe devolver el área del triángulo.***

function calcularAreaTriangulo(base, altura) {

// Fórmula para calcular el área de un triángulo: (base \* altura) / 2

return (base \* altura) / 2;

}

Explicación:

Parámetros:

base: La longitud de la base del triángulo.

altura: La altura del triángulo (la distancia perpendicular desde la base hasta el vértice opuesto).

Fórmula:

Se utiliza la fórmula estándar para calcular el área de un triángulo: (base \* altura) / 2.

Retorno:

La función devuelve el resultado del cálculo, que es el área del triángulo.

Ejemplo de uso:

let baseTriangulo = 10;

let alturaTriangulo = 5;

let area = calcularAreaTriangulo(baseTriangulo, alturaTriangulo);

console.log("El área del triángulo es:", area); // Imprimirá: "El área del triángulo es: 25"