Ejercicios de lógica

Objetivo

Los siguientes ejercicios tienen como objetivo practicar el pensamiento lógico en javascript para resolver problemas. Para la resolución de estos ejercicios dejamos a criterio la forma de resolución, se puede implementar una interacción con DOM para cargar / leer los datos o resolverlo por consola, lo importante es la resolución del problema. Luego, los ejercicios resueltos los pueden subir a un repositorio y enviar a molea@mobydigital.com y fmorel@mobydigital.com

Ejercicio 1: Contar instancias de un carácter en una cadena

- Cree una función que tome dos cadenas como argumentos y devuelva el número de veces que la primera cadena (el carácter único) se encuentra en la segunda cadena.

Ejemplo:

charCount("a", "edabit") → 1
charCount("c", "Chamber of secrets") → 1
charCount("b", "big fat bubble") → 4

Notas: Su salida debe distinguir entre mayúsculas y minúsculas (vea el segundo ejemplo).

Ejercicio 2: Encuentra los números más pequeños y más grandes

- Cree una función que tome una matriz de números y devuelva los números mínimo y máximo, en ese orden.

Ejemplo:

minMax([1, 2, 3, 4, 5]) → [1, 5]
minMax([2334454, 5]) → [5, 2334454]
minMax([1]) → [1, 1]

Notas: Todos los arrays de prueba tendrán al menos un elemento y son válidas.

Ejercicio 3: Dígito más alto

- Cree una función que tome un número como argumento y devuelva el dígito más alto de ese número.

Ejemplo:

sortDescending(123) → 321
sortDescending(1254859723) → 9875543221
sortDescending(73065) → 76530

Ejercicio 4: Ordenar números en orden descendente

- Cree una función que tome cualquier número no negativo como argumento y lo devuelva con sus dígitos en orden descendente. El orden descendente es cuando ordenas de mayor a menor.

Ejemplo:

sortDescending(123) → 321
sortDescending(1254859723) → 9875543221
sortDescending(73065) → 76530

Notas: Puede esperar números no negativos para todos los casos de prueba.

Ejercicio 5: Código postal válido

- Los códigos postales constan de 4 dígitos consecutivos. Dada una cadena, escriba una función para determinar si la entrada es un código postal válido. Un código postal válido es el siguiente:

- Solo debe contener números (no se permiten caracteres que no sean dígitos).
- No debe contener ningún espacio.
- No debe tener más de 5 dígitos de longitud.

Ejemplo:

isValid("1714") → true
isValid("12424") → false
isValid("732 2") → false
isValid("A323") → false

Ejercicio 6: Convertidor de temperatura

- Crea una función que convierta Celsius a Fahrenheit y viceversa.

Ejemplo:

convert("35ºC") → "95ºF"
convert("19ºF") → "-7ºC"
convert("33") → "Error"

Notas:

- Redondea al entero más cercano.
- Si la entrada es incorrecta, devuelve "Error".

Ejercicios 7: ¿Cuántos días hasta ...?

- Dado un date, devuelve cuántos días faltan para esa fecha. El date estará en formato mm/dd/aaaa .

Ejemplo:

daysUntil("10/16/2022") → "3 dias"
daysUntil("10/20/2020") → "7 dias"

Ejercicios 8: Reemplazar vocal con otro carácter

- Cree una función que tome una cadena y reemplace las vocales con otro carácter.

- A/a = 1
- E/e = 2
- I/i = 3
- O/o = 4
- U/u = 5

Ejemplo:

replaceVowel("karachi") → "k1r1ch3"
replaceVowel("chembur") → "ch2mb5r"
replaceVowel("khandbari") → "kh1ndb1r3"

Notas: La entrada puede estar en mayúsculas

Ejercicio 9: Devuelve la suma de los dos números más pequeños

- Cree una función que tome un array de números y devuelva la suma de los dos números positivos más bajos.

Ejemplos:

sumTwoSmallestNums([19, 5, 42, 2, 77]) → 7
sumTwoSmallestNums([10, 343445353, 3453445, 3453545353453]) → 3453455
sumTwoSmallestNums([2, 9, 6, -1]) → 8
sumTwoSmallestNums([879, 953, 694, -847, 342, 221, -91, -723, 791, -587]) → 563
sumTwoSmallestNums([3683, 2902, 3951, -475, 1617, -2385]) → 4519

Nota: No cuentes los números negativos.

Ejercicio 10: Calculadora básica

- Cree una función que tome dos números y un operador matemático + - / *y realice un cálculo con los números dados.

Ejemplo:

calculator(2, "+", 2) → 4
calculator(2, "*", 2) → 4
calculator(4, "/", 2) → 2

Notas: Tener en cuenta los posibles errores

¡Muchos éxitos!