

P4.1 EJERCICIOS BÁSICOS DE JAVASCRIPT

Índex

1. EJERCICIO 1	2
1.1. Solución	3
2. EJERCICIO 2	4
2.1. Solución	5
3. EJERCICIO 3	6
3.1. Solución	7
4. EXERCICI 4.....	8
4.1. Solución	9
5. EJERCICIO 5	10
5.1. Solución	11
6. EJERCICIO 6	12
6.1. Solución	13
7. EXERCICI 7.....	14
7.1. Solución	15
8. EJERCICIO 8	16
8.1. Solución	17
9. EJERCICIO 9	19
9.1. Solución	20

1. EJERCICIO 1

Partiendo de los ficheros adjuntos. Abra la carpeta en Visual Studio Code y visualiza el HTML y el archivo JS.

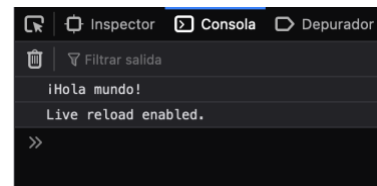
El primer ejercicio es realizar el enlace entre el fichero .HTML y el fichero .js.

Cuando lo haces, serás capaz de:

- Abrir la página con el navegador web (te recomiendo usar la extensión 'live server' para facilitar la recarga automática cuando se modifique algo de la web).
- Abrir el inspector del navegador web (F12).
- En el inspector, ver la consola y observar cómo tenemos nuestro primer "Hola mundo".

Página web de prueba

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Aspernatur, eligendi.



1.1.Solución

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <h1>Página web de prueba</h1>
  <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit.</p>

  <!--Incluir el enlace al javascript. -->
  <script src="javascript.js"></script>
</body>
</html>
```

2. EJERCICIO 2

Ahora debes hacer que la página enseñe al usuario un 'confirm' y si es positivo, en la consola se imprimra una "validación realizada" y si la confirmación es negativa se imprime "validación rechazada".

2.1.Solución

```
let result = confirm("¿Das tu validación?")
if (result) {
  console.log("Validación aceptada.")
} else {
  console.log("Validación rechazada.")
}
```

Ahora te muestro cómo se puede resolver este problema en una sola línea. No es necesario saber hacer este tipo de cosas, pero sí es común verlo si visitas código profesional.

```
console.log( confirm("¿Das tu validación?") ? "Validación aceptada." : "Validación rechazada.")
```

3. EJERCICIO 3

Ahora, debes hacer que la cuestión del ejercicio anterior se pregunte constantemente hasta obtener una validación. Aún así, cada vez que se haga la pregunta debe seguir mostrando lo mismo que antes por consola.

3.1.Solución

```
let result
do {
  result = confirm("¿Das tu validación?")
  if (result) {
    console.log("Validación aceptada.")
  } else {
    console.log("Validación rechazada.")
  }
} while (result == false)
```

4. EXERCICI 4

Borra el ejercicio anterior. Ahora tienes que solicitar dos números por pantalla con el 'prompt' (uno después de otro). Después, debes mostrar el resultado de la suma de estos números en la consola con un log.

4.1. Solución

```
let numA = Number(prompt("Primer Número"))  
let numB = Number(prompt("Segundo Número"))  
console.log(numA + numB)
```

5. EJERCICIO 5

El problema que tenemos con el ejercicio anterior es que el usuario podría introducir texto y no un número en el prompt. Modifícalo para que repitáis la petición hasta asegurarnos de que introdujo un número. Puedes utilizar la función "isNaN()".

Esta función devuelve un booleano dependiendo de si el valor que tiene en los paréntesis es un número o no. Tienes más información [aquí](#).

5.1.Solución

```
let numA
let numB

do {
  numA = Number(prompt("Primer número"))
} while (isNaN(numA))

do {
  numB = Number(prompt("Seugndo número"))
} while (isNaN(numB))

console.log(numA + numB)
```

6. EJERCICIO 6

Nuestra calculadora está un poco limitada, sólo sabe sumar. Haremos que pueda sumar, restar, multiplicar y dividir. Hazlo pidiendo en un tercer prompt la acción a realizar. Recuerda que el usuario se puede equivocar y no escribir bien la acción.

6.1.Solución

```
let numA
let numB
let verb

do {
  numA = Number(prompt("Primer número"))
} while (isNaN(numA))

do {
  numB = Number(prompt("Segundo número"))
} while (isNaN(numB))

do {
  verb = prompt("Deseas 'sumar', 'restar', 'multiplicar' o 'dividir'?")
} while (verb !== 'sumar' && verb !== 'restar' && verb !== 'multiplicar' && verb !== 'dividir')

let result = 0
switch (verb) {
  case 'sumar':
    result = numA + numB
    break
  case 'restar':
    result = numA - numB
    break
  case 'multiplicar':
    result = numA * numB
    break
  case 'dividir':
    result = numA / numB
    break
}

console.log(result)
```

7. EXERCICI 7

El problema de nuestra calculadora es que sólo puede realizar una operación. Así que tenemos que hacer que cuando acabe una operación, vuelva a iniciarse el proceso.

7.1.Solución

```
let numA
let numB
let verb

while (true) {
  do {
    numA = Number(prompt("Primer número"))
  } while (isNaN(numA))

  do {
    numB = Number(prompt("Segundo número"))
  } while (isNaN(numB))

  do {
    verb = prompt("Deseas 'sumar', 'restar', 'multiplicar' o 'dividir'?")
  } while (verb !== 'sumar' && verb !== 'restar' && verb !== 'multiplicar' && verb !== 'dividir')

  let result = 0
  switch (verb) {
    case 'sumar':
      result = numA + numB
      break
    case 'restar':
      result = numA - numB
      break
    case 'multiplicar':
      result = numA * numB
      break
    case 'dividir':
      result = numA / numB
      break
  }

  console.log(result)
}
```

8. EJERCICIO 8

Las calculadoras más básicas permiten realizar, es decir, operar con el último resultado obtenido. Sería de la siguiente forma:

1. El usuario entra en la página por primera vez.
2. El usuario introdujo el primer número que se le pide.
3. El usuario introduce el segundo número que se le pide.
4. El usuario introdujo la acción.
5. El usuario ve el resultado en la consola.
6. La página le está volviendo a pedir el primer número otra vez para realizar una operación.
7. El usuario en lugar de introducir un número, introdujo las letras 'ans'.
8. Este 'ans' quiere decir que el primer número es equivalente al resultado de la operación anterior (en el punto 4).
9. El usuario introduce el segundo número que se le pide.
10. El usuario introdujo la acción.
11. El usuario ve el resultado en la consola.

Puedes hacerlo primero para el primer número, pero después tienes que hacerlo para los dos números.

Nota: Ans es 0 si no hay acción anterior.

8.1.Solución

```
let numA
let numB
let verb
let ans = 0

while (true) {
  do {
    numA = prompt("Primer número")
    if (numA == 'ans') {
      numA = ans
    } else {
      numA = Number(numA)
    }
  } while (isNaN(numA))
  do {
    numB = prompt("Segundo número")
    if (numB == 'ans') {
      numB = ans
    } else {
      numB = Number(numB)
    }
  } while (isNaN(numB))
  do {
    verb = prompt("Deseas 'sumar', 'restar', 'multiplicar' o 'dividir'?")
  } while (verb != 'sumar' && verb != 'restar' && verb != 'multiplicar' && verb != 'dividir')

  let result = 0
  switch (verb) {
    case 'sumar':
      result = numA + numB
      break
    case 'restar':
      result = numA - numB
      break
    case 'multiplicar':
      result = numA * numB
      break
    case 'dividir':
```

```
    result = numA / numB  
    break  
}  
console.log(result)  
ans = result  
}
```

9. EJERCICIO 9

Te habrás dado cuenta de que no hemos usado funciones para nada y que estamos repitiendo código.

Especialmente nos ocurre con la petición de los números al inicio.

¿Podemos hacer una función que haga lo mismo y nos ahorre código? Hazlo sólo para la petición de números.

Nota: La función deberá:

- Obtener por valor de entrada el mensaje a mostrar en el pront.
- Obtener por valor de entrada el valor de ans.
- Devolver el número obtenido.

9.1.Solución

```
let numA
let numB
let verb
let ans = 0

function peticionNumeros(message, ans) {
  let num
  do {
    num = prompt(message)
    if (num == 'ans') {
      num = ans
    } else {
      num = Number(num)
    }
  } while (isNaN(num))

  return num
}

while (true) {
  numA = peticionNumeros("Primer número", ans)
  numB = peticionNumeros("Segundo número", ans)
  do {
    verb = prompt("Deseas 'sumar', 'restar', 'multiplicar' o 'dividir'?")
  } while (verb != 'sumar' && verb != 'restar' && verb != 'multiplicar' && verb != 'dividir')
  let result = 0
  switch (verb) {
    case 'sumar':
      result = numA + numB
      break
    case 'restar':
      result = numA - numB
      break
    case 'multiplicar':
      result = numA * numB
      break
    case 'dividir':
      result = numA / numB
  }
}
```

```
        break  
    }  
  
    console.log(result)  
    ans = result  
}
```