**¿Qué Devuelve Cada Función?**

**Recordá**: En la **Comunidad de Discord** podrás ver cómo otras personas tuvieron las mismas dudas y cuáles fueron las distintas formas en que las resolvieron. ¡Aprovechá los canales y las **clases gratuitas de Twitch** para revisar y consultar!

1. En este ejercicio deberás pensar qué devuelve cada Función sin ejecutarla en la consola:

function test1(x, y) {

return y - x

}

test1(10, 40)

**PISTA**

No importa el orden en que pasamos los **Argumentos** sino el de los **Parámetros** en el return.

function test2(x, y) {

return x \* 2

console.log(x)

return x / 2

}

test2(10)

**PISTA**

Cuando JS encuentra la *keyword* return, devuelve el valor pedido y termina la ejecución del bloque. Es decir, el resto del código que queda debajo no se ejecuta.

**Matemática Simple**

En este ejercicio deberás crear:

1. Una función que se llame triplicador que tome un número como *input* (osea, como Parámetro) y retorne el triple de ese valor.
2. Una función multiplicador que tome dos números como Parámetros y devuelva el producto de los dos.
3. Una función division que tome dos números como Parámetros y devuelva el resultado de dividir el primero por el segundo.
4. Una función resto que tome dos números como Parámetros y devuelva el resultado del módulo del primero sobre el segundo.

Por último, calculá el valor de triplicar 5, luego multiplicar eso por 12, dividir por 12 y encontrar el resto de dividir eso en 3.

⚠ **Importante:** Usá solamente las funciones que escribiste antes, sin otros operadores.

### contarDeA\_n

En este ejercicio deberás escribir una Función llamada contarDeA\_n que tenga los Parámetros contar\_de\_a y contar\_hasta. Además, deberá escribir en la consola los números desde el 1 hasta contar\_hasta en intervalos de contar\_de\_a.

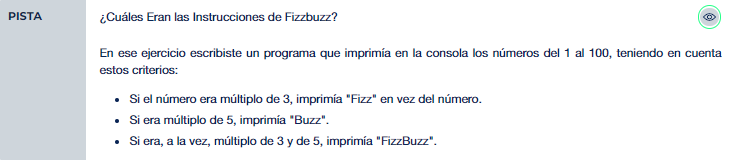
**PISTA**

Si ponemos 2 y 10 como Argumentos en el llamado, la Función deberá contar de a dos hasta llegar a diez.

### Desafío ***FizzBuzz II***

Se trata del **DESAFÍO** de la clase que te proponemos que lo compartas en DRIVE cuando termines.

En este ejercicio, deberás escribir una nueva versión de FizzBuzz (fizzBuzz2) que tome dos *Strings* como Argumento y reimplemente el FizzBuzz original. Elegí una palabra para cada *String* (palabra1 y palabra2) que reemplace a Fizz y a Buzz.



Para completar este ejercicio, deberás:

1. Lograr que fizzBuzz2 devuelva un *String* con los números separados por comas.
2. Mejorar la Función para que el usuario pueda decidir hasta qué número tiene que contar fizzBuzz2.
3. Mejorar la Función para que el usuario pueda ingresar fizz\_num y buzz\_num para que la sustitución de palabras ocurra en los números múltiplos de los nuevos argumentos de entrada (en vez de solo 3 y 5).

⚠**Importante:** Intentá no ayudarte con tu código anterior de Fizzbuzz.