## Ejercicios básicos con funciones:

1. Crear una función que al invocar escriba por pantalla: Hola grupo!!
2. Crear una función que al invocar escriba por consola: Esto es un log de errores
3. Crear una función que reciba un parámetro y muestre por pantalla si el número es par o impar, utilizar una función.
4. Crear una función llamada “suma” que reciba 2 parámetros y los sume: muestra el resultado con un alert dentro de la función.
5. Crear una función llamada “resta” que reciba 2 parámetros y los reste: muestra el resultado con un alert dentro de la función.
6. Crear una función llamada “multiplica” que reciba 2 parámetros y los multiplique: muestra el resultado con un alert dentro de la función.
7. Crear una función llamada “divide” que reciba 2 parámetros y los divida: muestra el resultado con un alert dentro de la función.
8. Crear una función que permita calcular el cubo de un número: muestra el resultado con un alert dentro de la función.
9. Crear una función que muestre un saludo por pantalla. La función recibirá por pantalla el nombre y mostrará el texto “Hola XXXX” : muestra el resultado con un alert dentro de la función.
10. Crear una función que muestre por pantalla los números del 1 hasta el número que reciba como parámetro: muestra el resultado con document.write o console.log

El número final se solicitará al usuario antes de invocar a la función.

1. Crear una función que pida un nombre, un apellido y un lugar y a raíz de ahí muestre en pantalla una pequeña historia: muestra el resultado con document.write
2. Realizar un programa que, mediante una función denominada permuta, implemente la permutación de los valores de dos variables enteras que se rellenaron en la función principal.

## Ejercicios básicos funciones con retorno:

1. Una función que reciba cinco números enteros como parámetros y muestre por pantalla el resultado de sumar los cinco números (hay que devolver el valor y mostrar el resultado fuera de la función).
2. Una función que reciba cinco números enteros como parámetros y muestre por pantalla el resultado de sumar los cinco números (hay que devolver el valor y mostrar el resultado fuera de la función).

Asigna el resultado de una invocación (llamada) a la función con parámetros de entrada, por ejemplo:

2, 5, 1, 8, 10

a una variable que tenga de nombre “temporal”, multiplica esa variable por 2 y muestra por pantalla el valor de la variable en pantalla.

1. Una función que reciba como parámetros el valor del radio y la altura de un cilindro y devuelva el volumen del cilindro, teniendo en cuenta que el volumen de un cilindro se calcula como Volumen = númeroPi \* radio \* radio \* Altura siendo númeroPi = 3.1416 aproximadamente. Muestra por pantalla el valor de la variable en pantalla.
2. Crea un script donde declares una función obtenerImporteConImpuestos que reciba dos parámetros: el importe sin impuestos (numérico) y el tipo de producto (numérico entero). La función debe mostrar por pantalla el importe sin impuestos más el 21% si el tipo de producto es 1, ó el importe sin impuestos más el 10% si el tipo de producto es 2, ó el importe sin impuestos más el 5% si el tipo de producto es 3.

Ejemplo: obtenerImporteConImpuestos(100, 1) debe mostrar: Para un importe sin impuestos de 100 y tipo de producto 1 el resultado de importe con impuestos es 121. obtenerImporteConImpuestos(100, 2) debe mostrar: Para un importe sin impuestos de 100 y tipo de producto 2 el resultado de importe con impuestos es 110.

1. Crea un script donde declares una función obtenerImporteConImpuestos2 que reciba un parámetro: el importe sin impuestos (numérico). La función debe devolver un array con valor undefined para el índice 0, el importe sin impuestos más el 21% para el índice 1, el importe sin impuestos más el 10% para el índice 2, ó el importe sin impuestos más el 5% para el índice 3. Invoca la función haciendo que se muestre el contenido del array por pantalla.

obtenerImporteConImpuestos(100) debe devolver: resultado[0] = undefined, resultado[1] = 121, resultado[2] = 110, resultado[3] = 105. Por pantalla se debe mostrar: Para precio sin impuestos 100 si el producto es tipo 1 el importe es 121, si el producto es tipo 2 el importe es 110 y si el producto es tipo 3 el importe es 105.

1. Crea una función que reciba como parámetro un array de 5 posiciones, dentro de la función rellena el array con números aleatorios y devuelve el array para finalmente imprimir por consola. Nota: la impresión debe hacerse fuera de la función.
2. Crear una función que reciba un array multidimensional de 3x3, dentro de la función rellena el array con número aleatorios. El array relleno debe enviarse a otra función como parámetro de entrada, donde se recorrerá para mostrar la tabla de 3x3 en pantalla (con document.write)
3. Se debe implementar una función que al recibir dos números A y B, indique la cantidad N que hay que sumarle a A para que A+N sea divisible entre B. Por ejemplo, si A=4 y B=3, N será 2.
4. Se debe implementar un programa en que permite jugar al juego Piedra, Papel, Tijera contra la máquina.

El juego costará de varias rondas y ganará el que gane 2 rondas. El juego terminará cuando el usuario ya no quiera seguir jugando. El programa deberá mostrar en cada ronda, las rondas que lleva ganadas cada jugador. Y al final de la partida, las partidas que lleva ganadas cada jugador.

Para implementar el juego, además de la función principal, como mínimo, el programa debe contener las siguientes funciones:

● Función “apuestaUsuario”. Este módulo debe pedir al usuario si quiere Piedra, Papel o Tijera. (Piedra (p), Papel(a) o Tijera(t)).

● Función “piedraPapelTijera”. Este módulo mostrará y devolverá si la máquina ha

elegido Piedra, Papel o Tijera. Debe calcularlo de manera aleatoria, para ello debe

utilizar la función random(). Por lo ejemplo, la función devolverá Piedra si “valor” es 0, devolverá Papel si “valor” es 1 o devolverá Tijera si “valor” es 2.

● Función “comprobar”. Este módulo comprobará cuál de los dos jugadores ha ganado la ronda y mostrará los resultados. También comprobará si alguno de los dos

jugadores ha ganado ya 2 rondas, y por tanto, la partida. Si, es así, mostrará quién ha ganado y las partidas que lleva ganadas cada jugador

10. Crea una aplicación que nos pida un número por teclado y con un método se lo pasamos por parámetro para que nos indique si es o no un número primo, debe devolver **true** si es primo sino **false**.

Un número primo es aquel solo puede dividirse entre 1 y si mismo. Por ejemplo: 25 no es primo, ya que 25 es divisible entre 5, sin embargo, 17 si es primo.