

Ejercicios rápidos

Funciones

1. Crear la función **suma** que acepte como argumentos **dos números** y devuelva **el resultado de su suma**
2. Crear la función **potenciacion** que acepte como argumentos **dos números** y devuelva **el resultado de elevar el primero al segundo**
3. Crear la función **separarPalabras** que acepte como argumento **un string** que contenga más de una palabra y devuelva **un vector de palabras, por ejemplo: 'hola mundo' => c(hola, mundo)**
4. Crear la función **repetirString** que acepte como argumentos **un string y un número** y devuelva **un string que sea el resultado de concatenar el primer string el número dado de veces**
5. Crear la función **esPrimo** que acepte como argumento **un número menor que 42** y devuelva **True si es primo y False si no lo es o si es mayor o igual que 42**
6. Crear la función **multiplicacion** que acepte como argumentos **dos números** y devuelva **el resultado de su multiplicación**
7. Crear la función **division** que acepte como argumentos **dos números** y devuelva **el resultado de su division**
8. Crear la función **esPar** que acepte como argumento **un número** y devuelva **True si es par y False si es impar**
9. Crear la función **ordenarLista** que acepte como argumento **un vector de números** y devuelva un **vector ordenado de menor a mayor**
10. Crear la función **obtenerPares** que acepte como argumento **un vector de números** y devuelva un **vector con los elementos pares**
11. Crear la función **pintarVector** que acepte como argumento **un vector** y devuelva **una cadena de texto, por ejemplo: c(1, 2, 3) => '(1, 2, 3)'**
12. Crear la función **eliminarDuplicados** que acepte como argumento **un vector** y devuelva **un vector en el que se hayan eliminado los duplicados**
13. Crear la función **ordenarVector2** que acepte como argumento **un vector de números** y devuelva un **vector ordenado de mayor a menor**
14. Crear la función **obtenerImpares** que acepte como argumento **un vector de números** y devuelva un **vector con los elementos impares**
15. Crear la función **sumarLista** que acepte como argumento **un vector de números** y devuelva **la suma de los números en el vector, por ejemplo: c(1, 2, 3) => 6**
16. Crear la función **multiplicarVector** que acepte como argumento **un vector de números** y devuelva **la multiplicación de los números en el vector, por ejemplo: c(2, 3, 4) => 24**
17. Crear la función **multiplicacionPro** que acepte como argumentos **dos números** y devuelva **el resultado de su multiplicación**, pero NO PUEDE usar el operador de multiplicación (*) para calcularlo ni usar otras funciones realizadas previamente. Es decir, solo puede usar el operador suma (+)