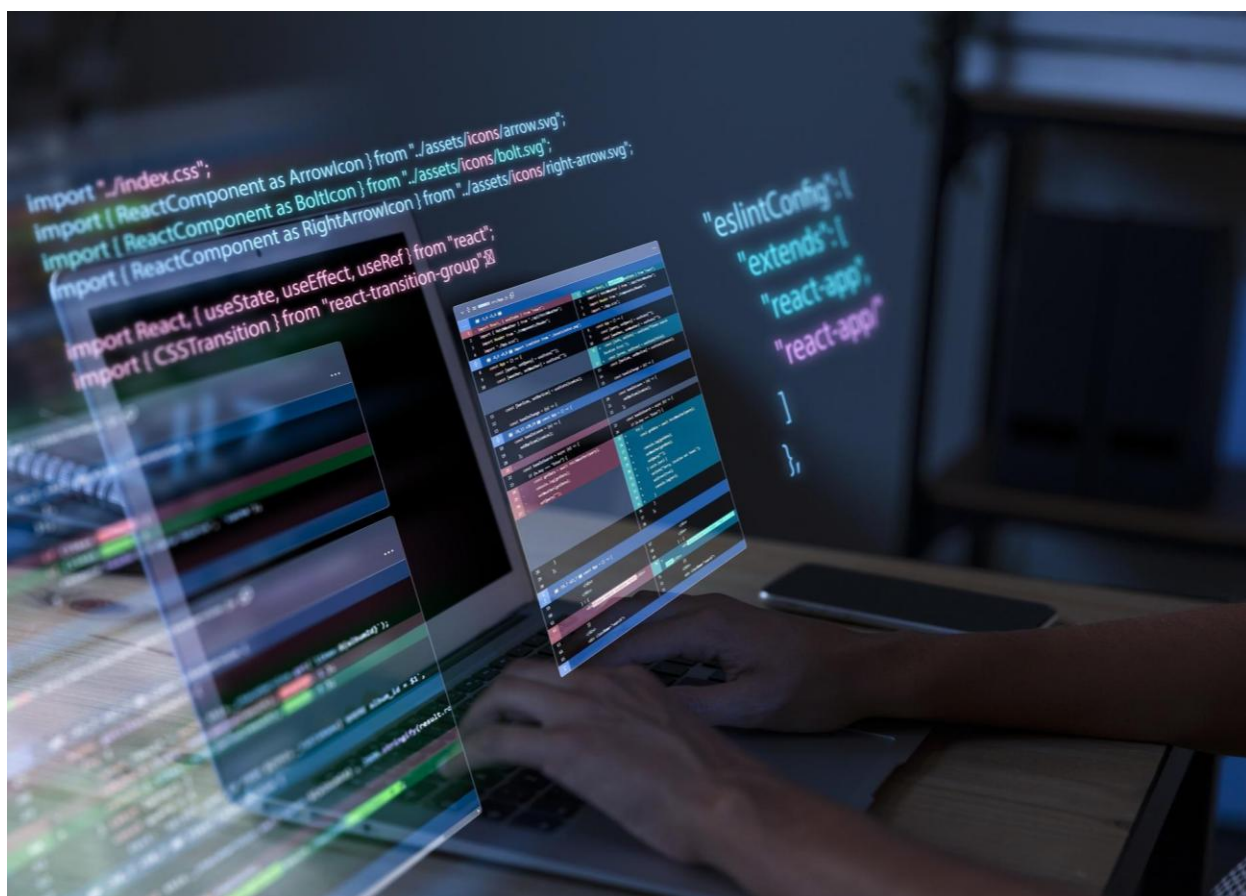


DAM-DAW/PROGRAMACIÓN

# Ejercicios

## Funciones (II)

---



**Autor:** Marcela Martín

**Fecha:** noviembre 2025

1. Diseña una función a la que se le pasan las horas y minutos de dos instantes de tiempo, con el siguiente prototipo:

*static int diferenciaMin(int hora1, int minuto1, int hora2, int minuto2)*

La función devolverá la cantidad de minutos que existen de diferencia entre los dos instantes utilizados.

2. Crea un programa que muestre por consola una serie de números aleatorios enteros.

Utiliza una función que requiere los siguientes parámetros:

- cantidad de números aleatorios que se mostrarán y
- los valores mínimos y máximos que estos pueden tomar.

Valida que la cantidad de números es un número mayor o igual a cero.

Utiliza una función para la validación, que devuelva un valor booleano.

3. Sobrecarga la función creada en el ejercicio anterior con un único parámetro que indicará la cantidad de números aleatorios a crear (sin rango), en ese caso, serán números reales aleatorios entre 0 y 1.
4. Escribe un programa que calcule la operación de potencia (base elevado a exponente) utilizando una función a la que se le pasen ambos valores, solicitados previamente al usuario. Los valores deben ser números enteros. Utiliza un bucle para obtener el resultado.
5. Realiza un programa que resuelva el mismo problema que el anterior ejercicio, pero utilizando la recursividad.
6. Escribe un programa que cuente las vocales de una cadena de texto introducida por el usuario. Utiliza una función (contarVocales()) que tenga como parámetro una cadena de texto y devuelva un entero.

El programa solicita nuevas cadenas de texto hasta que el usuario introduce 'salir'.

Ver métodos `length()` y `charAt()` de la clase `String`.

7. Escribe un programa que busque en una cadena de texto dada por el usuario un determinado carácter. Se solicita al usuario la cadena de texto y el carácter a buscar. Utiliza una función (`buscaChar()`) que tenga como parámetro cadena y carácter a buscar y devuelva 'verdadero' si se ha localizado el carácter. Falso en caso contrario.

Desde el programa principal se imprimirá como resultado el mensaje "El carácter X se encuentra en la cadena ZZZZ" Siendo X y ZZZZ los datos de entrada.

8. Modifica el programa anterior para que la función de búsqueda retorne la posición en la que se encuentra la primera coincidencia del carácter buscado (`buscaPosChar()`). Si no se encuentra, retorna -1

Ejemplo: Buscar la posición del carácter O en la cadena HOLA : devuelve 2.

9. Escribe un programa que solicite una palabra al usuario e imprima la misma cadena en orden inverso, por ejemplo 'HOLA' se imprime como 'ALOH'. Utiliza una función (`invertirCadena`) que acepte la cadena como parámetro y devuelva la cadena invertida.