

# Programación de servicios y procesos – Manual Ejercicio 1 RA01

Juan Francisco Mateo

## Programación de Procesos y Servicios



# Programación de servicios y procesos – Manual Ejercicio 1 RA01

Juan Francisco Mateo

## Proyecto GeneraCadenas

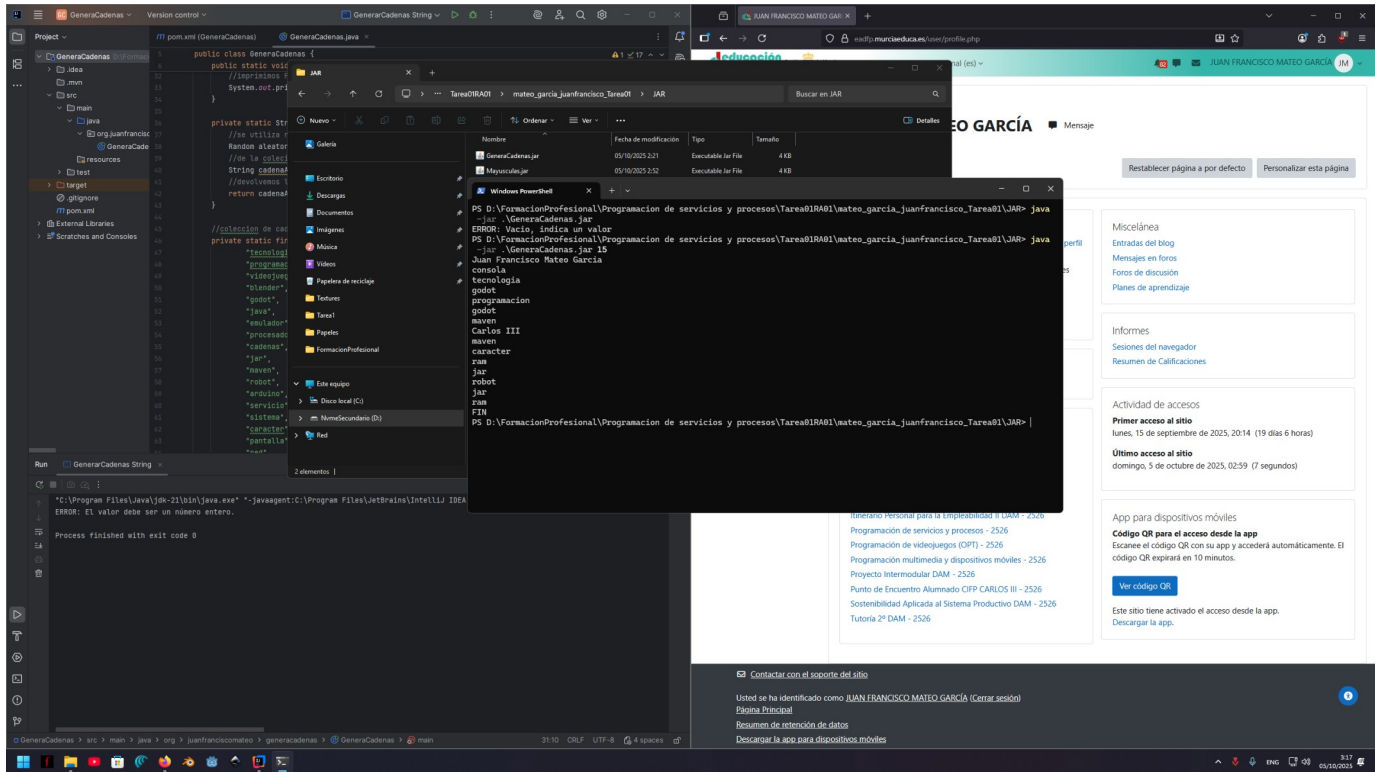
Para ejecutar este programa necesitamos pasar obligatoriamente el numero de cadenas que queremos por argumentos, de lo contrario nos arroja un error

The image shows two side-by-side screenshots. The left screenshot displays the IntelliJ IDEA IDE with the 'GeneraCadenas' project open. The code in 'GeneraCadenas.java' includes a main method that takes an array of arguments, a method to generate a random number, and a collection of category names like 'tecnologia', 'programacion', 'videojuegos', etc. The right screenshot shows the user profile page for 'JUAN FRANCISCO MATEO GARCÍA' on the 'murciaeduca.es' website. The page includes navigation links, user details (email, zone), course details (DAM - 2526), and a QR code for mobile app access.

This image is similar to the one above but shows an error state. On the left, the IntelliJ IDEA IDE shows the same code, but the 'Run' output at the bottom displays the message: 'ERROR: El valor debe ser un numero entero.' (Error: The value must be an integer). The right screenshot remains the same, showing the user profile page on 'murciaeduca.es'.

# Programación de servicios y procesos – Manual Ejercicio 1 RA01

Juan Francisco Mateo



En la primera captura y segunda, usando la documentación proporcionada, he creado parámetros de ejecución para cuando el usuario introduce una string (Provocando una excepción) y cuando el usuario introduce un numero de ejemplo, en este caso 10.

En la ultima captura podemos ver como el programa funciona en la terminal, y si no se proporcionan argumentos este lanza otro error.



# Programación de servicios y procesos – Manual Ejercicio 1 RA01

Juan Francisco Mateo

## Proyecto Mayusculas

☞ Cuando ejecutamos el programa de manera aislada, podemos escribir cadenas de texto y estas serán almacenadas en memoria para luego ser convertidas a mayusculas.

The screenshot displays two side-by-side windows. The left window is an IDE (IntelliJ IDEA) showing the source code of the 'Mayusculas' program. The code is in Java and uses a Scanner to read input from the user, converting it to uppercase. The right window is a web browser showing the user profile page for 'JUAN FRANCISCO MATEO GARCÍA' on the 'educación digital' website. The page includes navigation links, user details, and a list of courses.

```
package org.juanfrancisco.mayusculas;

import java.util.Scanner;
import java.util.ArrayList;
import java.util.List;

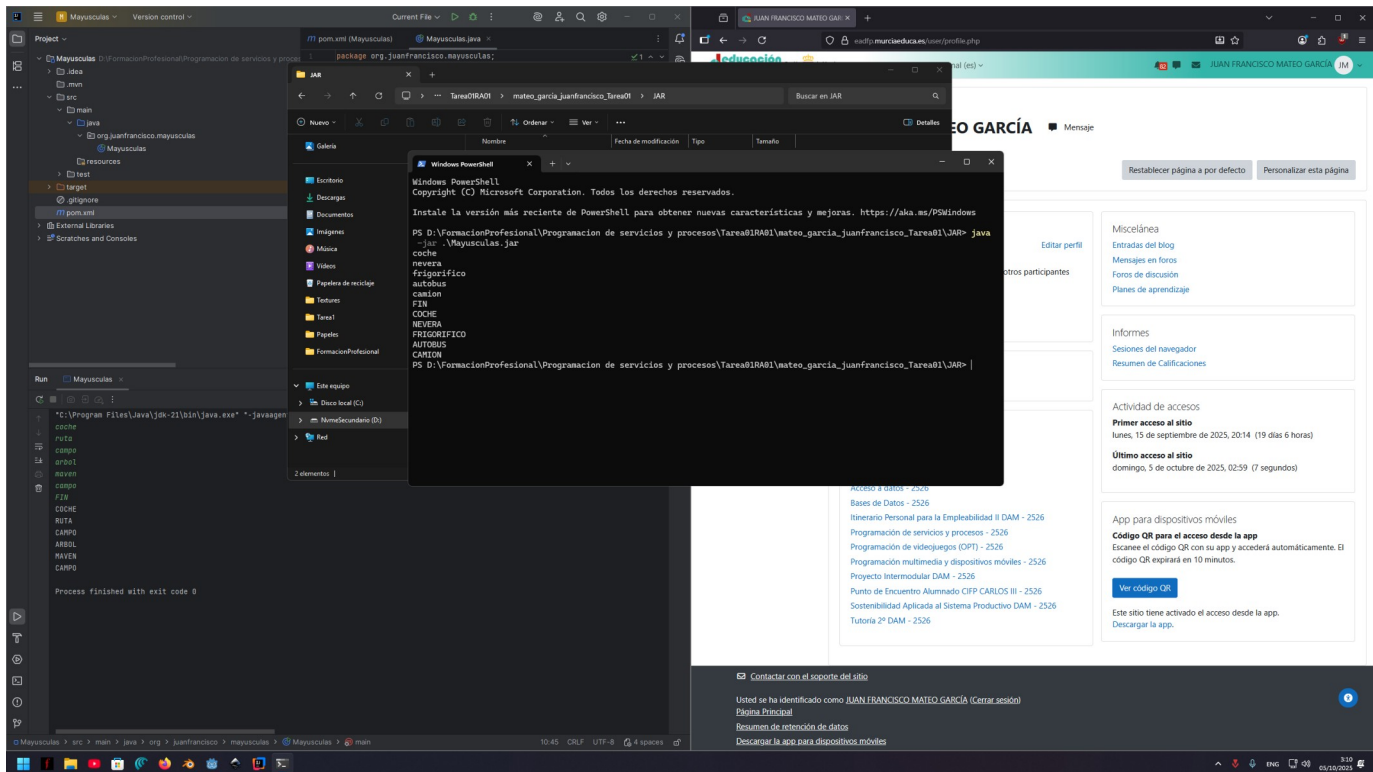
public class Mayusculas {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        List<String> palabras = new ArrayList<>();
        //Introducción hasta escribir "FIN"
        while (sc.hasNextLine()) {
            //almacenamos la cadena del usuario
            String cadena = sc.nextLine();
            if (cadena.equals("FIN")) {
                //si la palabra es FIN no guardamos esta y cortamos
                break;
            } else {
                //de lo contrario la guardamos en la colección
                palabras.add(cadena);
            }
        }
        //cerramos scanner
        sc.close();
    }
}
```

This screenshot is identical to the one above, showing the same IDE and web browser windows. The IDE window shows the same Java code for the 'Mayusculas' program. The web browser window shows the same user profile page for 'JUAN FRANCISCO MATEO GARCÍA'.

# Programación de servicios y procesos – Manual Ejercicio 1 RA01

Juan Francisco Mateo



En la primera captura podemos observar como al ejecutar el programa dentro del entorno de IntelliJ podemos escribir la serie de cadenas que queremos, separadas por un salto de linea (Pulsando la “tecla Enter”).

Cuando no queramos que obtenga mas valores, escribiremos FIN y convertirá a mayúsculas las cadenas que le hemos introducido.(Segunda captura)

En la tercera y ultima captura, podemos comprobar que el programa funciona (después de compilarlo en un .jar) al ejecutarlo dentro de un entorno java.

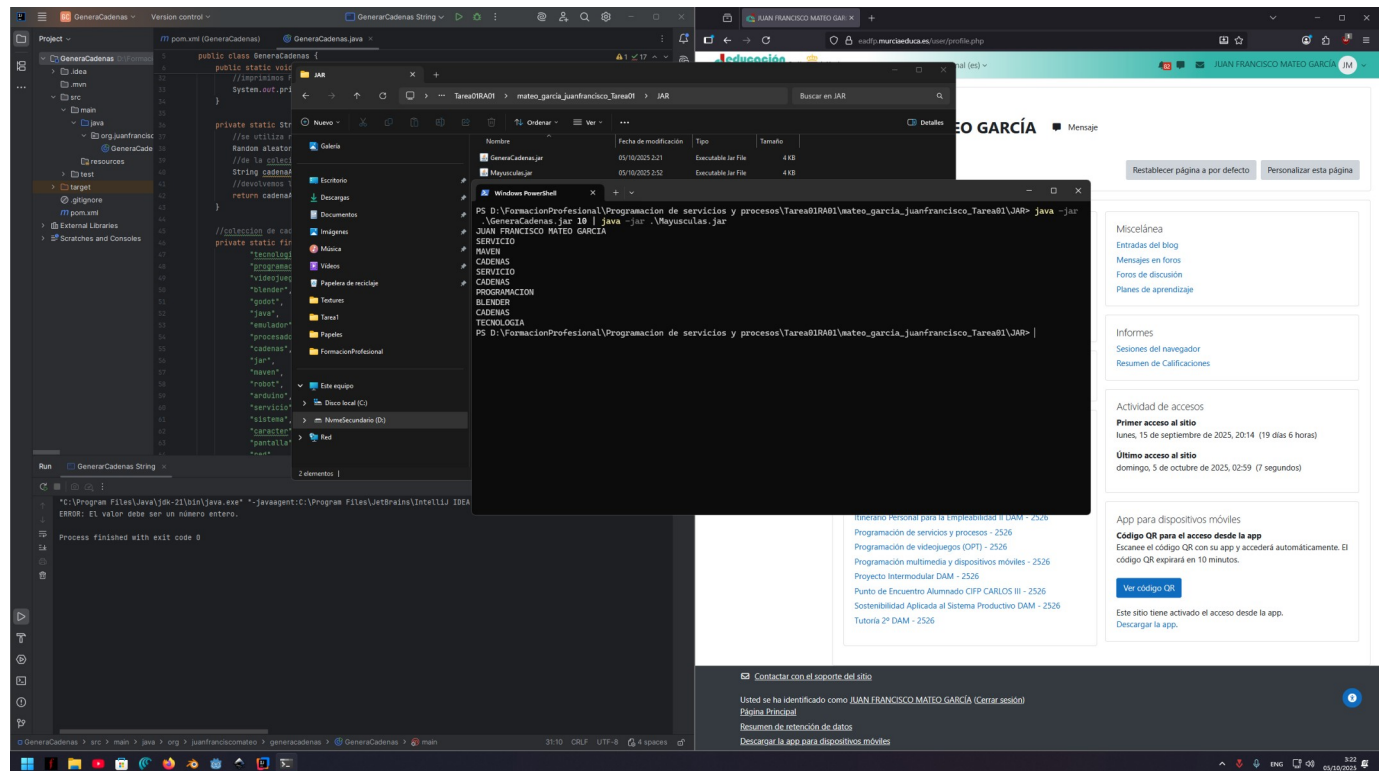
# Programación de servicios y procesos – Manual Ejercicio 1 RA01

Juan Francisco Mateo

## Túnel entre ambos

Como podemos comprobar, podemos usar ambos programas con un tunel, usando el siguiente comando

```
java -jar .\GeneraCadenas.jar 10 | java -jar .\Mayusculas.jar
```



Y este nos genera con el numero que le digamos, una lista de cadenas aleatorias y luego las convierte todas a mayusculas.