

Actividad Práctica - Tema 4:

Acceso Aleatorio a Ficheros

Objetivo

Utilizar la clase `RandomAccessFile` para escribir y leer registros de texto desde posiciones específicas de un archivo. Aprender a manipular el puntero del archivo mediante `seek()` y a controlar la lectura secuencial o puntual de datos.

Instrucciones

1. Crea una carpeta llamada `datos` en tu proyecto.
2. Crea una clase Java llamada `AccesoAleatorioEjercicio`.
3. Implementa un programa que:

Escriba tres registros con texto en un archivo binario usando `writeUTF`.

Lea y muestre el primer y segundo registro usando `seek` y `readUTF`.

Mida y muestre la posición del puntero antes y después de cada operación.

```
import java.io.*;

public class AccesoAleatorioEjercicio {

    public static void main(String[] args) {

        try {

            // 1. Crear archivo binario y escribir tres
registros

            RandomAccessFile raf = new
RandomAccessFile("datos/registros.dat", "rw");

            raf.writeUTF("Registro 1");

            raf.writeUTF("Registro 2");

            raf.writeUTF("Registro 3");
```

```
        // 2. Volver al inicio para leer

        raf.seek(0);

        // Leer primer registro

        System.out.println("Posición antes de leer 1: " +
raf.getFilePointer());

        String r1 = raf.readUTF();

        System.out.println("Registro 1: " + r1);

        System.out.println("Posición después de leer 1: "
+ raf.getFilePointer());

        // Leer segundo registro

        System.out.println("Posición antes de leer 2: " +
raf.getFilePointer());

        String r2 = raf.readUTF();

        System.out.println("Registro 2: " + r2);

        System.out.println("Posición después de leer 2: "
+ raf.getFilePointer());

        raf.close();

    } catch (IOException e) {

        System.out.println("Error: " + e.getMessage());

    }

}
```

```
}
```

```
/*
```

Preguntas de reflexión con respuestas:

1. ¿Qué indica el valor que devuelve `getFilePointer()`?

- Indica la posición actual del puntero dentro del archivo, es decir, el byte donde se encuentra listo para leer o escribir.

2. ¿Qué sucede si cambias el orden de lectura?

- Puedo leer los registros en cualquier orden usando `seek()`. Si leo en orden diferente, solo necesito mover el puntero a la posición correcta, de lo contrario leería datos equivocados.

3. ¿Por qué `RandomAccessFile` no es recomendable para archivos de texto plano sin estructura?

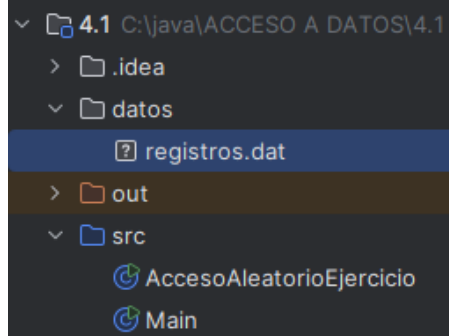
- Porque `RandomAccessFile` funciona mejor con archivos con registros de tamaño conocido o estructurado. En texto plano, los registros pueden tener longitudes variables, lo que hace difícil calcular posiciones y manipular datos de forma segura.

4. ¿Cómo podrías modificar solo el tercer registro sin afectar los demás?

- Primero uso `seek()` para mover el puntero al inicio del tercer registro, y luego uso `writeUTF()` para sobrescribirlo. Así, los otros registros permanecen intactos.

```
*/
```

```
"C:\Program Files\Java\jdk-21\bin\java.exe" "-  
Posición antes de leer 1: 0  
Registro 1: Registro 1  
Posición después de leer 1: 12  
Posición antes de leer 2: 12  
Registro 2: Registro 2  
Posición después de leer 2: 24  
  
Process finished with exit code 0
```



4.1 C:\java\ACCESO A DATOS\4.1

- > .idea
- ▼ datos
 - registros.dat
- > out
- ▼ src
 - AccesoAleatorioEjercicio
 - Main