

Actividad UF1-1.

Lanzamiento de procesos. Triángulo numérico.

Programación de Servicios y
procesos

Ivan Nuñez Rodriguez

2ºDAM

8/10/204

Índice:

1. Enunciado
2. Código.
3. Desarrollo.
 - a. Triangulo.
 - b. LanzadorTringulo.
4. Resultado.

1. Enunciado

Esta práctica consiste en el lanzamiento de varios procesos, cada uno de los cuales genera un triángulo numérico.

- Se trata de un programa de consola, que requiere que el usuario introduzca un número como argumento y avisa en caso de que no sea introducido.
- Crea otro programa Java que inicie tres procesos simultáneos lanzando tres veces el programa *Triangulo* con los parámetros 5, 7 y 9.
- La salida de los tres procesos debe ser enviada a los archivos *triangulo5.txt*, *triangulo7.txt* y *triangulo9.txt*.
- Al principio de cada fichero de salida y antes del triángulo, debe escribirse la fecha de inicio del proceso. Para ello, tendrás que modificar la clase *Triangulo*.
- Al final de cada fichero de salida, después del triángulo debes escribir la fecha de finalización del proceso. Para ello, tendrás que modificar la clase *Triangulo*.

2. Código:

```
import java.time.LocalDateTime;

public class Triangulo {
    public static void main(String[] args) {
        //se introduce fecha de entrada
        System.out.println("Fecha y hora de inicio: " + LocalDateTime.now());
        if (args.length == 0) {
            System.out.println("Se requiere un argumento");
            return;
        }
        int filas = Integer.parseInt(args[0]);
        for (int i = filas; i >= 1; i--) {
            for (int n = 1; n <= i; n++) {
                System.out.print(n);
            }
            System.out.println();
        }
        //se introduce fecha de salida
        System.out.println("Fecha y hora de salida: " + LocalDateTime.now());
    }
}
```

```

import java.io.File;
import java.io.IOException;

public class LanzadorTriangulo {
    public static void main(String[] args) {
        try {
            //se crea classpath para ubicar archivos
            String classpath = System.getProperty("java.class.path");
            //se crean procesos con argumento
            ProcessBuilder triangulo5 = new ProcessBuilder("java", "-cp", classpath,
"Triangulo", String.valueOf(5));
            ProcessBuilder triangulo7 = new ProcessBuilder("java", "-cp", classpath,
"Triangulo", String.valueOf(7));
            ProcessBuilder triangulo9 = new ProcessBuilder("java", "-cp", classpath,
"Triangulo", String.valueOf(9));
            //se crean salidas
            triangulo5.redirectOutput(new File("src/triangulo5.txt"));
            triangulo7.redirectOutput(new File("src/triangulo7.txt"));
            triangulo9.redirectOutput(new File("src/triangulo9.txt"));
            // se si inician procesos
            triangulo5.start();
            triangulo7.start();
            triangulo9.start();
            //se recoge exception
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}

```

3. Desarrollo.

a. Triangulo.

Se copia el código dispuesto en el ejercicio y se añaden los LocalDateTime() al principio y final de la ejecución tal como se pide en el ejercicio.

b. LanzadorTringulo.

Se crean ProcessBuilder() tantos como pide el ejercicio en este caso 3 se introducen como argumentos los solicitados en este caso 5,7,9 llamando al proceso "Triangulo".

Y se redirige la salida con redirecoutput() como se pide el ejercicio en este caso en la misma ubicación que los archivos java.

Como se requiere que la ejecución sea a la vez se inicializan al mismo tiempo con. star().

Se crea una excepción de tipo IOException que mandara un mensaje en el caso de fallo de entrada salida.

4. Resultado.

a. Trinangulo5.txt.

```
Fecha y hora de inicio: 2024-10-08T13:22:52.755064400
12345
1234
123
12
1
Fecha y hora de salida: 2024-10-08T13:22:52.764877300
```

b. Trinangulo7.txt.

```
Fecha y hora de inicio: 2024-10-08T13:22:52.771057400
1234567
123456
12345
1234
123
12
1
Fecha y hora de salida: 2024-10-08T13:22:52.778669700
```

c. Triangulo9.txt.

```
Fecha y hora de inicio: 2024-10-08T13:22:52.775157500
123456789
12345678
1234567
123456
12345
1234
123
12
1
Fecha y hora de salida: 2024-10-08T13:22:52.781544800
```