

Ej. Procesos Aleatorios. Java. Solución

Aleatorios.java

```
import java.util.*;
public class aleatorios {
    public static void main(String Args[]){
        try {
            int numero = new Random().nextInt(49)+1;
            System.out.println(numero);
        } catch (Exception e) { e.printStackTrace(); } } }
```

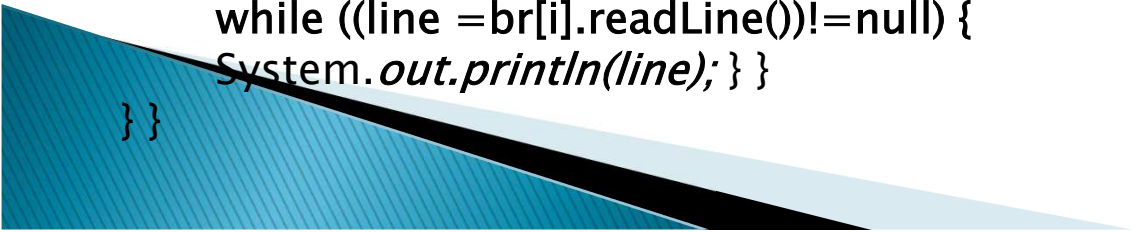
MAIN.JAVA

```
import java.io.*;
public class main {
    public static void main(String[] args){
        try{
            final int loteria = 6;
            String [] numeros = new String [loteria];
            for(int i=0; i<loteria; i++){
                numeros[i]=(new BufferedReader(new InputStreamReader(new
                ProcessBuilder("java", "-jar",
                "aleatorios.jar").start().getInputStream()))).readLine(); }

            for(int i=0; i<loteria; i++){ System.out.println(numeros[i]);}
        } catch (Exception e){ e.printStackTrace(); }}
```

Procesos Aleatorios. Solución

```
import java.io.BufferedReader; import java.io.IOException;
import java.io.InputStream; import java.io.InputStreamReader;
import java.io.*; import java.util.Arrays;
public class comunicacionentreprocesos_Hijosaleatorios {
static int naleatorios=6;
public static void main(String args[])throws IOException,InterruptedException {
int i;
Process[] p = new Process[naleatorios]; //Array de procesos
InputStream is []= new InputStream[naleatorios];
InputStreamReader isr []= new InputStreamReader[naleatorios];
BufferedReader br [] = new BufferedReader[naleatorios];
for(i=0;i<naleatorios;i++)      {
    p[i]= new ProcessBuilder("..\\aleatorio").start();
    is[i]= p[i].getInputStream();
    isr[i]= new InputStreamReader(is[i]);
    br[i] = new BufferedReader(isr[i]);    }
for(i=0;i<naleatorios;i++)      {
    p[i].waitFor();    }
String line;
for(i=0;i<naleatorios;i++){
    System.out.println("Salida del proceso Aleatorio "+ i + ":");
    while ((line =br[i].readLine())!=null) {
        System.out.println(line); } }
}}
```



Prácticas Procesos

- ▶ Supóngase que en el pc existe una carpeta Usuarios que contiene una subcarpeta por cada uno de ellos, desde Usuario1 hasta Usuario5. A su vez, cada carpeta contiene al menos dos subcarpetas y ficheros en cada una de ellas.
- ▶ Crear un programa en java que cree varios procesos en paralelo.
- ▶ Cada uno de ellos debe listar el contenido de la carpeta del usuario que recibe como argumento y escribir en un fichero de salida los archivos listados.
- ▶ El final de la ejecución devuelve un fichero por cada carpeta de usuario.
- ▶ Una vez finalizados todos los hijos el padre imprimirá los ficheros por pantalla secuencialmente.

Memorias. ESQUEMA DE MEMORIA

- ▶ Introducción
 - Antecedentes
 - Entorno de Desarrollo
 - ▶ Enunciado Práctico
 - Explicación del Problema a resolver
 - ▶ Solución
 - Descripción y Justificación de la solución.
 - Código fuente de la Aplicación e Interfaces
 - ▶ Explicación de
 - Algoritmos
 - Clases
 - Estructuras de Datos
 - ▶ Listado de pruebas
 - Entrada
 - Ejecución
 - Salida
 - ▶ Conclusiones
- 