

EJERCICIOS UNIDAD 1

2. RA1 -CE f) Sincronización de procesos en Java

La clase Process nos permite tomar el control de la ejecución del proceso usando:

- `p.waitFor()`:
 - El proceso Padre espera a que el hijo “p” finalice o termine.
- `p.waitFor(long timeout, TimeUnit unit)`:
 - El proceso padre espera “timeout” o hasta que el proceso hijo “p” finaliza o termina.
 - Documentación de TimeUnit:
<https://docs.oracle.com/en/java/javase/18/docs/api/java.base/java/util/concurrent/TimeUnit.html>
- `p.exitValue()`:
 - Retorna el valor de salida del proceso hijo.
 - El proceso hijo debe haber terminado antes de llamar a este método si no se **lanzarla la excepción `IllegalThreadStateException`**.
- Otra forma de hacer esperar a un proceso es con `Thread.sleep(int milliseconds)`, que nos permite pausar el proceso actual.

Ejercicios de sincronización de procesos

6. Crea un programa en Java que imprima ¿qué pasa?, 10 veces esperando un segundo cada vez.

7. Crea un programa igual el ejercicio 3, pero justo después de crear el proceso hijo el padre espera a que termine antes de generar los números aleatorios. ¿Qué ocurre? Por último, obtén el valor de salida.

8. En este ejercicio el estudiante debe crear un programa que:

- Abra una aplicación introducida por argumento (`String[] args`), por ejemplo “**Notepad**” en Windows o “**gedit**” en Linux.
- 10 segundos después se cerrará el proceso (con código no a mano)
- Después de que la app se cierre, el programa espera 5 segundos y se imprimirá “Buenas noches amigo!”.