

# A3: Crear un Hilo Implementando la Interfaz Runnable

---

## 1. Objetivo de Aprendizaje:

El objetivo de esta actividad es que los alumnos comprendan cómo crear hilos en Java utilizando la **interfaz Runnable**, en lugar de extender la clase `Thread`. A través de la simulación de una carrera de corredores, similar a la actividad anterior, los estudiantes aprenderán a implementar la interfaz `Runnable` para separar la lógica de la tarea del control del hilo. Esto ayuda a reforzar la flexibilidad de la programación multihilo en Java.

---

## 2. Enunciado:

Crea una clase `Corredor` que implemente la interfaz `Runnable` para simular una carrera de corredores. Debes realizar lo siguiente:

1. Define la clase `Corredor` que implemente la interfaz `Runnable`.
2. Cada corredor debe tener un nombre, y en el método `run()` debe imprimirse un mensaje indicando el progreso del corredor.
3. Crea tres instancias de la clase `Corredor` y utiliza la clase `Thread` para lanzar los hilos que ejecutarán la carrera.
4. Lanza los tres hilos para que compitan en la carrera de forma concurrente.
5. Al finalizar, imprime un mensaje que indique que la carrera ha terminado.

## Actividad3

```
1 package actividad;
2
3 public class Actividad3 implements Runnable {
4     private String nombre;
5
6     public Actividad3(String nombre) {
7         this.nombre = nombre;
8     }
9
10    @Override
11    public void run() {
12        for (int i = 1; i <= 10; i++) {
13            System.out.println(nombre + " ha avanzado " + i + " metros.");
14            try {
15                Thread.sleep((int) (Math.random() * 1000));
16            } catch (InterruptedException e) {
17                System.out.println(nombre + " ha sido interrumpido.");
18            }
19        }
20        System.out.println(nombre + " ha terminado la carrera.");
21    }
22
23    public static void main(String[] args) {
24        Actividad3 corredor1 = new Actividad3("Corredor 1");
25        Actividad3 corredor2 = new Actividad3("Corredor 2");
26        Actividad3 corredor3 = new Actividad3("Corredor 3");
27
28        Thread hilo1 = new Thread(corredor1);
29        Thread hilo2 = new Thread(corredor2);
30        Thread hilo3 = new Thread(corredor3);
31
32        hilo1.start();
33        hilo2.start();
34        hilo3.start();
35
36        try {
37            hilo1.join();
38            hilo2.join();
39            hilo3.join();
40        } catch (InterruptedException e) {
41            System.out.println("La carrera ha sido interrumpida.");
42        }
43
44        System.out.println("La carrera ha terminado.");
45    }
46 }
```

## Resultado

```
Corredor 1 ha avanzado 1 metros.  
Corredor 2 ha avanzado 1 metros.  
Corredor 3 ha avanzado 1 metros.  
Corredor 2 ha avanzado 2 metros.  
Corredor 3 ha avanzado 2 metros.  
Corredor 1 ha avanzado 2 metros.  
Corredor 2 ha avanzado 3 metros.  
Corredor 3 ha avanzado 3 metros.  
Corredor 1 ha avanzado 3 metros.  
Corredor 3 ha avanzado 4 metros.  
Corredor 1 ha avanzado 4 metros.  
Corredor 2 ha avanzado 4 metros.  
Corredor 1 ha avanzado 5 metros.  
Corredor 3 ha avanzado 5 metros.  
Corredor 3 ha avanzado 6 metros.  
Corredor 2 ha avanzado 5 metros.  
Corredor 2 ha avanzado 6 metros.  
Corredor 3 ha avanzado 7 metros.  
Corredor 1 ha avanzado 6 metros.  
Corredor 1 ha avanzado 7 metros.  
Corredor 3 ha avanzado 8 metros.  
Corredor 2 ha avanzado 7 metros.  
Corredor 3 ha avanzado 9 metros.  
Corredor 1 ha avanzado 8 metros.  
Corredor 1 ha avanzado 9 metros.  
Corredor 3 ha avanzado 10 metros.  
Corredor 3 ha terminado la carrera.  
Corredor 2 ha avanzado 8 metros.  
Corredor 1 ha avanzado 10 metros.  
Corredor 2 ha avanzado 9 metros.  
Corredor 1 ha terminado la carrera.  
Corredor 2 ha avanzado 10 metros.  
Corredor 2 ha terminado la carrera.  
La carrera ha terminado.
```