

### Actividad 2.7.1

Crea un programa Java que lance cinco hilos, cada uno incrementara una variable contador de tipo entero, compartida por todos, 5000 veces y luego saldrá. Comprobar el resultado final de la variable. ¿Se obtiene el resultado correcto? Ahora sincroniza el acceso a dicha variable.

Lanza los hilos primero mediante la clase Thread y luego mediante el interfaz Runnable. Comprueba el resultado.

Código con Thread:

- Código HiloA:

```
public class HiloA extends Thread {
    //instanciamos la clase contador en el hilo
    private Contador contador;
    //Le añadimos parámetros
    public HiloA(String n, Contador c) {
        setName(n);
        contador = c;
    }

    public void run() {
        //Ponemos en el synchronized el objeto
        synchronized (contador) {
            for (int j = 0; j < 5000; j++) {
                contador.incrementa();
            }
            System.out.println(getName() + " contador vale " + contador.valor());
        }
    }
}
```

- Clase Contador:

```
public class Contador {

    private int c; //atributo contador

    public Contador() {
        this.c = 0;
    }
    //Suma la variable contador
    public void incrementa() {
        c = c + 1;
    }

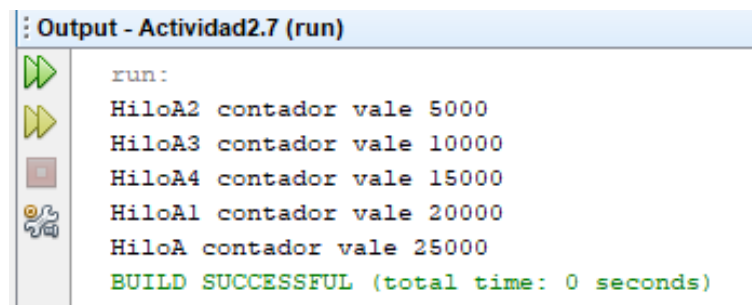
    //Devuelve la variable del contador
    public int valor() {
        return c;
    }
}
```

```
}
```

- Código Main:

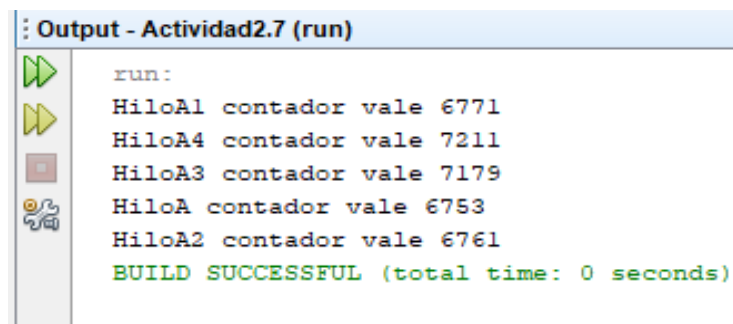
```
public class Main {  
    public static void main(String[] args){  
        //En la clase Contador ya inicializo a 0 la variable Contador que es la que tienen en  
        comun los 5 hilos  
        Contador cont = new Contador();  
        //Inicializo los hilos  
        HiloA a = new HiloA("HiloA", cont);  
        HiloA a1 = new HiloA("HiloA1", cont);  
        HiloA a2 = new HiloA("HiloA2", cont);  
        HiloA a3 = new HiloA("HiloA3", cont);  
        HiloA a4 = new HiloA("HiloA4", cont);  
  
        a.start();  
  
        a1.start();  
  
        a2.start();  
  
        a3.start();  
  
        a4.start();  
  
    }  
}
```

Con Synchronized sale lo siguiente:



```
Output - Actividad2.7 (run)  
run:  
HiloA2 contador vale 5000  
HiloA3 contador vale 10000  
HiloA4 contador vale 15000  
HiloA1 contador vale 20000  
HiloA contador vale 25000  
BUILD SUCCESSFUL (total time: 0 seconds)
```

Sin Synchronized sale lo siguiente= van cambiando los valores aleatoriamente:



```
Output - Actividad2.7 (run)  
run:  
HiloA1 contador vale 6771  
HiloA4 contador vale 7211  
HiloA3 contador vale 7179  
HiloA contador vale 6753  
HiloA2 contador vale 6761  
BUILD SUCCESSFUL (total time: 0 seconds)
```

Código con Runnable:

- Clase Main:

```
public class Actividad27_Runnable {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        //En la clase Contador ya inicializo a 0 la variable Contador que es la que tienen en
        comun los 5 hilos
        Contador cont = new Contador();
        //Inicializo los hilos
        Thread hilo1=new Thread(new Hilo("Primer Ejercicio",cont));
        Thread hilo2=new Thread(new Hilo("Primer Ejercicio",cont));
        Thread hilo3=new Thread(new Hilo("Primer Ejercicio",cont));
        Thread hilo4=new Thread(new Hilo("Primer Ejercicio",cont));
        Thread hilo5=new Thread(new Hilo("Primer Ejercicio",cont));

        hilo1.start();
        hilo2.start();
        hilo3.start();
        hilo4.start();
        hilo5.start();

    }
}
```

- Clase Contador:

```
public class Contador {

    private int c; //atributo contador

    public Contador() {
        this.c = 0;
    }
    //Suma la variable contador
    public void incrementa() {
        c = c + 1;
    }

    //Devuelve la variable del contador
    public int valor() {
        return c;
    }
}
```

- Clase Hilo:

```
public class Hilo implements Runnable {
```

```
//instanciamosla clase contador en el hilo
private Contador contador;
public String nombre;

//Le añadimos parámetros
public Hilo(String nombre, Contador c) {
    this.nombre = nombre;
    contador = c;
}

public void run() {
    //Ponemos en el synchronized el objeto

    for (int j = 0; j < 5000; j++) {
        contador.incrementa();
    }
    System.out.println(nombre + " contador vale " + contador.valor());
}
}
```