

### Actividad 1.1

Crea dos clases (hilos) Java que extiendan la clase Thread. Uno de los hilos debe visualizar en pantalla en un bucle infinito la palabra TIC y el otro hilo la palabra TAC. Dentro del bucle utiliza el método sleep() para que nos de tiempo a ver las palabras que se visualizan cuando lo ejecutemos, tendrás que añadir un bloque try-catch (para capturar la excepción InterruptedException). Crea después la función main() que haga uso de los hilos anteriores. ¿Se visualizan los textos TIC y TAC de forma ordenada (es decir TIC TAC TIC TAC ...)?

De forma desordenada se muestra en la consola.

He hecho dos clases una llamada hilo1 y Proceso1.

```
package hilo1;

/**
 *
 * @author david
 */
public class Hilo1 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) throws InterruptedException {
        Proceso1 hilo1 = new Proceso1("TIC");
        Proceso1 hilo2 = new Proceso1("TAC");
        hilo1.start();
        hilo2.start();
    }
}
```

```
package hilo1;
```

```
import java.util.logging.Level;
```

```
import java.util.logging.Logger;
```

```
/**
```

```
 *
```

```
 * @author david
```

```
 */
```

```
public class Proceso1 extends Thread{
```

```
    private String cadena;
```

```
    public Proceso1(String cadena){
```

```
        this.cadena=cadena;
```

```
    }
```

```
    @Override
```

```
    public void run(){
```

```
        while(true){
```

```
            System.out.println(cadena);
```

```
            try{
```

```
                Thread.sleep(500);
```

```
            } catch (InterruptedException ex) {
```

```
                Logger.getLogger(Proceso1.class.getName()).log(Level.SEVERE, null, ex);
```

```
            }
```

```
        }
```

```
    }
```

```
}
```

Output - Actividad2.1 (run)

TAC  
TIC  
TAC  
TIC  
TAC  
TIC  
TIC  
TAC  
TAC  
TIC  
TIC  
TAC  
TIC  
TAC  
|