

Clase de kotlin donde vimos el uso y manejo de los hilos en android

```
class MainActivity : AppCompatActivity() {

    var contador = 0
    var estado = true
    var pausa = false

    override fun onCreate(savedInstanceState: Bundle?) {
        var hilo=Hilo(this)
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val btnIniciar=findViewById<Button>(R.id.btnIniciar)
        val btnPausar=findViewById<Button>(R.id.btnPausar)
        val btnContinuar=findViewById<Button>(R.id.btnContinuar)
        val btnReiniciar=findViewById<Button>(R.id.btnReiniciar)
        val btnDetener=findViewById<Button>(R.id.btnDetener)

        btnIniciar.setOnClickListener {
            hilo.start()
        }

        btnPausar.setOnClickListener {
            pausa=true
        }

        btnContinuar.setOnClickListener {
            pausa=false
        }

        btnReiniciar.setOnClickListener {
            contador=0
        }

        btnDetener.setOnClickListener {
            if(hilo.isAlive){
                estado=false
                return@setOnClickListener
            }
        }

    }

    class Hilo(activity:MainActivity):Thread(){
        var act=activity
```

```
override fun run() {  
    super.run()  
    while (act.estado){  
        while (act.pausa==true){  
            sleep(100)  
        }  
        sleep(100)  
        act.runOnUiThread {  
            act.tvNumero.setText("Hilo: "+act.contador)  
        }  
        act.contador++  
    }  
}  
}
```

hiloFactorial

INICIAR

PAUSAR

CONTINUAR

REINICIAR

DETENER

Hilo: 16