

T2.2. Recursos y estados de los hilos

Recursos compartidos por los hilos.

Cuando creamos varios objetos de una clase, puede ocurrir que varios hilos de ejecución accedan a un objeto. Es importante recordar que **todos los campos del objeto son compartidos entre todos los hilos**.

Supongamos una clase como esta:

```
public class Empleado(){
    int numHorasTrabajadas=0;
    public void incrementarHoras(){
        numHorasTrabajadas++;
    }
}
```

Si varios hilos ejecutan sin querer el método `incrementar` en teoría debería ocurrir que el número se incrementase tantas veces como procesos.

Estados de un hilo. Cambios de estado.

Aunque no lo vemos, un hilo cambia de estado: puede pasar de la nada a la ejecución. De la ejecución al estado «en espera». De ahí puede volver a estar en ejecución. De cualquier estado se puede pasar al estado «finalizado».

El programador no necesita controlar esto, lo hace el sistema operativo. Sin embargo un programa multihilo mal hecho puede dar lugar problemas como los siguientes:

- Interbloqueo. Se produce cuando las peticiones y las esperas se entrelazan de forma que ningún proceso puede avanzar.
- Inanición. Ningún proceso consigue hacer ninguna tarea útil y por lo tanto hay que esperar a que el administrador del sistema detecte el interbloqueo y mate procesos (o hasta que alguien reinicie el equipo).