

# Cas T6

1) Tenim una base de dades relacional, que descriu un model simplificat d'un banc, amb les següents taules i les seves corresponents columnes:

Taula "clientes"		
Columna	Tipus	Descripció
cod_cliente	integer unique not null	Codi de client
tipo_doc	char(1) not null	"N"=NIF, "E"=NIE, "P"=passaport
num_doc	char(16) not null	Nombre del document identificador
cod_pais	char(2) not null	Codi de dues lletres del país ("ES"=Espanya)
apellidos	char(40) not null	Cognoms
nombre	char(40) not null	Nom(s) de pila

Taula "cuentas"		
Columna	Tipus	Descripció
num_cuenta	char(20) unique not null	Nombre de compte
saldo	decimal(10,2) not null	Saldo actual en euros
interes	decimal (10,8) not null	Tipus d'interès diari
sucursal	decimal(4) not null	Codi de sucursal

Taula "movimientos"		
Columna	Tipus	Descripció
cuenta	char(10) not null	Codi de compte
fecha	timestamp not null	Data i hora de l'operació
importe	decimal(10,2) not null	Import de l'operació (pot ser positiu o negatiu)
texto	char(40) not null	Text descriptiu

Taula "intervinientes"		
Columna	Tipus	Descripció
cliente	integer not null	Codi del client intervinent
cuenta	char(10) not null	Nombre de compte del client
tipo	char(1) not null	Tipus intervinent: "T"=titular, "A"=autoritzat

2) Disposam del següent codi Java:

```
class MyRunnable implements Runnable {

    private boolean doStop = false;

    public synchronized void doStop() {
        this.doStop = true;
    }

    private synchronized boolean keepRunning() {
        return this.doStop == false;
    }

    public void run() {
        while(keepRunning()) {
            System.out.println("Running");

            try {
                Thread.sleep(3 * 1000); //milliseconds
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}

public class MyRunnableMain {

    public static void main(String[] args) {
        MyRunnable myRunnable = new MyRunnable();

        Thread thread = new Thread(myRunnable);

        thread.start();

        try {
            Thread.sleep(10 * 1000); //milliseconds
        } catch (InterruptedException e) {
            e.printStackTrace();
        }

        myRunnable.doStop();
    }
}
```

Respondre a les següents preguntes sobre el codi:

a) En executar el mètode MyRunnableMain.main, quin resultat es mostrarà en pantalla?  
(Valor: 10%)

b) Quant de temps aproximadament durarà l'execució en condicions normals? Per què?  
(Valor: 10%)

c) Què passaria si en el codi original substituïm la línia "Thread.sleep(3 \* 1000);" per  
"Thread.sleep(300);"? (Valor: 15%)

d) Què passaria si en el codi original substituïm la línia "while(keepRunning()) {" per la  
següent? (requeriria a més un "import java.time.\*" al principi). (Valor: 15%)

```
while(keepRunning() && (LocalDateTime.now().getHour() < 18)) {
```

Valor de la pregunta: 50% de la nota del cas