

# POOII – Ex Threads II

ContaBancaria.java

```
public class ContaBancaria {
    private int saldo;

    ContaBancaria()
    {
        saldo = 1000;
    }
    public void sacar()
    {
        saldo -= 200;
    }
    public void depositar()
    {
        saldo += 200;
    }
    public int getSaldo() {
        return saldo;
    }
}
```

Pessoa.java

```
public class Pessoa implements Runnable{
    Thread thread;
    ContaBancaria cb = new ContaBancaria();

    public Pessoa(String name)
    {
        thread = new Thread(this, name);
        thread.start();
    }

    @Override
    public void run()
    {
        System.out.println(thread.getName() + " Iniciada.\n");
        for(int i = 0; i < 10; i++)
        {
            try{
                synchronized (cb) {
                    cb.depositar();
                    System.out.println("Saldo de " + thread.getName() + " após depósito: " +
cb.getSaldo());
                }
                Thread.sleep(200);
                synchronized (cb) {
                    cb.sacar();
                    System.out.println("Saldo de " + thread.getName() + " após saque: " +
cb.getSaldo());
                }
                Thread.sleep(200);
            } catch (InterruptedException exc) {
                System.out.println("Thread Interrompida");
            }
        }
        System.out.println(thread.getName() + " Finalizada. Saldo Final: " + cb.getSaldo());
    }
}
```

UsaBanco.java

```
public class UsaConta {  
    public static void main(String[] args) {  
        Pessoa p1 = new Pessoa("Pessoa #1");  
        Pessoa p2 = new Pessoa("Pessoa #2");  
  
        try{  
            p1.thread.join();  
            p2.thread.join();  
        } catch (InterruptedException exc) {  
            System.out.println("Thread Interrompida");  
        }  
    }  
}
```