

# Atividade 1 - Programação concorrente em Java

## Exercício 1: Minha Primeira Thread

Neste exercício, você irá criar e executar sua primeira thread em Java. Você criará uma thread que imprime números de 1 a 10 com um intervalo de 1 segundo entre eles.

### Objetivos:

1. Crie uma classe `MinhaPrimeiraThread` que estenda a classe `Thread`.
2. No método `run` da classe `MinhaPrimeiraThread`, faça um loop de 1 a 10 e imprima o número atual do loop.
3. Entre cada iteração do loop, faça a thread dormir por 1 segundo.
4. No método `main`, crie uma instância de `MinhaPrimeiraThread` e a inicie.

### Desafio Adicional:

1. Crie outra classe `MinhaSegundaThread` que também estenda a classe `Thread`.
2. `MinhaSegundaThread` deve imprimir os números de 10 a 1 (decrescente) com um intervalo de 1 segundo entre eles.
3. Inicie `MinhaSegundaThread` juntamente com `MinhaPrimeiraThread` no método `main`.

## Exercício 2: Simulação de Conta Bancária

### Descrição:

Neste exercício, você vai simular uma conta bancária básica com operações básicas de depósito, saque e consulta de saldo, em um ambiente *multithreaded*.

### Objetivos:

1. Crie uma classe `ContaBancaria` que recebe um saldo inicial pelo construtor.
2. A classe `ContaBancaria` deve ter métodos `depositar`, `sacar` e `getSaldo` que alteram ou consultam o saldo da conta.
3. Instancie um objeto do tipo `ContaBancaria` com um saldo inicial de R\$ 1000.
4. Crie duas threads: uma que realiza 100 depósitos de R\$ 200 cada e outra que realiza 100 saques de R\$ 200 cada.
5. Ao final, imprima o saldo final da conta. O resultado esperado é R\$ 1000.

### Desafio Adicional:

1. Utilize um bloco `synchronized` para garantir que as operações na conta bancária sejam *thread-safe*.