

# Threads

- Fios de execução concorrentes de um programa
- Um processo tem uma ou mais threads
- Partilham recursos e comunicam entre si através de memória partilhada
- Analogia:
  - As threads de um processo são como vários cozinheiros que seguem as instruções do mesmo livro de culinária, mas não necessariamente todos na mesma página.

# Threads em Java

- `java.lang.Runnable`
  - interface implementada por classes cujas instâncias representam threads
  - classes que implementem esta interface têm que implementar o método `run()`
- `java.lang.Thread`
  - implementa `java.lang.Runnable`
  - classes que estendam `Thread` devem re-implementar o método `run()`
  - outros métodos relevantes: `start()`, `sleep()`, `join()`

## Exemplo HelloWorld:

```
public class HelloWorld implements Runnable {  
    public void run() {  
        System.out.println("Hello from a thread!");  
    }  
  
    public static void main(String args[]) {  
        (new Thread(new HelloWorld())).start();  
    }  
}
```

## Exemplo HelloWorld 2:

```
public class HelloRunnable2 implements Runnable {
    int n;

    public void run() {
        System.out.println(n);
    }

    HelloRunnable2(int a) {
        n=a;
    }

    public static void main(String args[]) {
        HelloRunnable2 r222 = new HelloRunnable2(222);
        HelloRunnable2 r111 = new HelloRunnable2(111);
        Thread t1=new Thread(r222);
        Thread t2=new Thread(r111);
        System.out.println("Antes");
        t1.start();
        t2.start();
        System.out.println("Depois");
        try {
            t2.join();
            t1.join();
        } catch (InterruptedException e) {}
        System.out.println("Fim");
    }
}
```

## Exemplo HelloWorld 3:

```
public class HelloRunnable3 implements Runnable {
    int n;
    public void run() {
        System.out.println(n);
        this.set(111);
    }
    HelloRunnable3(int a) {
        n=a;
    }

    public void set(int b) { n=b; }

    public static void main(String args[]) {
        HelloRunnable3 r=new HelloRunnable3(222);
        Thread t1=new Thread(r);
        Thread t2=new Thread(r);
        System.out.println("Antes");
        t1.start();
        t2.start();
        System.out.println("Depois");
        try {
            t2.join();
            t1.join();
        } catch (InterruptedException e) {}
    }
}
```

## ● Exemplo HelloWorld 3:

T<sub>1</sub>

print(n)

n=111

T<sub>2</sub>

print(n)

n=111

Output:

222

111

## ● Exemplo HelloWorld 3:

$T_1$

`print(n)`

`n=111`

$T_2$

`print(n)`

`n=111`

Output:

222

222