

# Threads

## Capítulo XI

## Threads

- Muitas vezes, deseja-se executar tarefas de forma simultânea, ou seja, ao mesmo tempo;
- Para programas separados, o próprio Sistema Operacional faz o gerenciamento por meio de processos paralelos. Dessa forma, utiliza-se com eficiência as várias CPUs existentes.
- No caso de um único programa, isso é feito por meio de **Threads** .

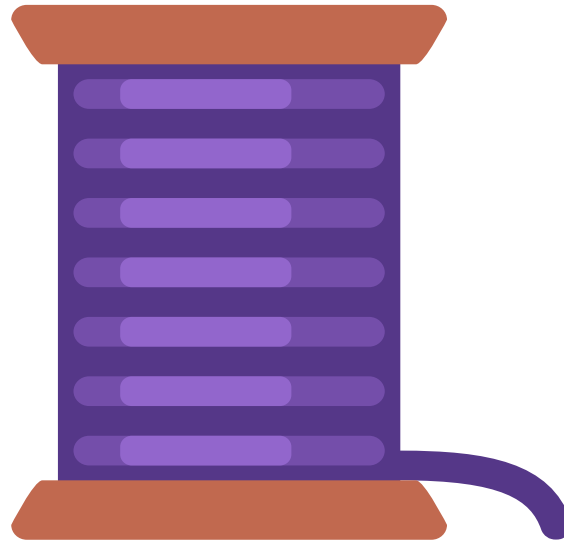


## Threads

- **Thread** é uma forma como uma tarefa de um programa é dividido em duas ou mais tarefas que podem ser executadas simultaneamente.
- Uma thread permite, por exemplo, que o utilizador de um programa utilize uma funcionalidade do ambiente enquanto outras linhas de execução realizam outros cálculos e operações.
- Em equipamentos que possuem apenas uma única CPU, cada thread é processada de forma aparentemente simultânea, pois a mudança entre uma thread e outra é feita de forma tão rápida que para o utilizador, isso está acontecendo concorrentemente. Em equipamentos com múltiplos CPUs, multi-cores, as threads são realizadas realmente de forma simultânea.

## Implementação

- Existem duas maneiras de se realizar a utilização de Thread no Java. As duas formas são:
  - Utilizando Herança (Classe Thread);
  - Utilizando a Interface Runnable;



## Implementação

- Nessa aula, iremos usar como exemplo o jogo Fortnite;
- Fortnite é um jogo multiplayer, o qual possui um mapa e os jogadores devem sobreviver durante a partida. Dessa maneira, existem algumas tarefas que devem ser executadas em paralelo. Entre elas, podemos citar:
  - Gerar o Mapa;
  - Tocar o som ambiente;



## Herdando da Classe Thread

- Para fazer a herança em ambas as classes, faremos da seguinte forma:

```
public class SomAmbiente extends Thread{  
    @Override  
    public void run() {  
    }  
}  
  
public class Mapa extends Thread{  
    @Override  
    public void run() {  
    }  
}
```

A classe Thread implementa a interface *Runnable*. Portanto, podemos sobrescrever o método para o que desejamos fazer especificamente.

## Especificando os Métodos

- Classe SomAmbiente:

```
@Override
public void run() {
    for (int i = 0; i < 3; i++) {

        System.out.println("Tocando a música!");

        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```

## Especificando os Métodos

- Classe Mapa:

```
@Override
public void run() {
    for (int i = 0; i < 3; i++) {

        System.out.println("Gerando o mapa!");

        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```



## Classe ExemploThread

- Na classe ExemploThread, em seu método main, iremos realizar as instâncias e iniciar cada uma das threads:

```
public class ExemploThread {  
    public static void main(String[] args) {  
        // Realizando as instâncias  
        SomAmbiente somAmbiente = new SomAmbiente();  
        Mapa mapa = new Mapa();  
  
        // Iniciando as Threads  
        somAmbiente.start();  
        mapa.start();  
    }  
}
```

## Resultados

Tocando a música!

Gerando o mapa!

Tocando a música!

Gerando o mapa!

Tocando a música!

Gerando o mapa!

## Implementando a Interface Runnable

- Para fazer a implementação em ambas as classes, faremos da seguinte forma:

```
public class SomAmbiente implements Runnable{  
    @Override  
    public void run() {  
    }  
}  
  
public class Mapa implements Runnable{  
    @Override  
    public void run() {  
    }  
}
```

## Especificando os Métodos

- Classe Mapa:

```
@Override
public void run() {
    for (int i = 0; i < 3; i++) {

        System.out.println("Gerando o mapa!");

        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```

## Especificando os Métodos

- Classe Mapa:

```
@Override
public void run() {
    for (int i = 0; i < 3; i++) {

        System.out.println("Gerando o mapa!");

        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```

## Classe ExemploThread

- Na classe ExemploThread, em seu método main, iremos realizar as instâncias e iniciar cada uma das threads:

```
public class ExemploThread {  
    public static void main(String[] args) {  
        // Realizando as instâncias  
        SomAmbiente somAmbiente = new SomAmbiente();  
        Mapa mapa = new Mapa();  
  
        // Instanciando as threads  
        Thread threadSomAmbiente = new Thread(somAmbiente);  
        Thread threadMapa = new Thread(mapa);  
  
        // Iniciando as threads  
        threadSomAmbiente.start();  
        threadMapa.start();  
    }  
}
```

## Resultados

```
Tocando a música!  
Gerando o mapa!  
Tocando a música!  
Gerando o mapa!  
Tocando a música!  
Gerando o mapa!
```



**Obrigado!**