

Cada processo possui diversas threads (linhas de instruções), assim nós podemos dividir partes do nosso processo (programa em Java) para trabalhar paralelamente.z

Em um **programa em Java podemos querer executar 2 ou mais threads ao mesmo tempo, ou seja, 2 ou mais procedimentos internos do programa ao mesmo tempo.**

perceba aqui claramente que temos um procedimento que pode demorar minutos, e não vai retornar nada ao usuário, apenas fazer comunicação interna entre Web Services. Não podemos parar toda nossa aplicação para executar o procedimento . Pois imagina se o usuário está fazendo um cadastro simples e tem que esperar 4 minutos para terminar o processamento .

A solução então é fazer com que esse procedimento seja executado concorrentemente, ou seja, ao mesmo tempo em que o usuário está realizando o cadastro.

Para criar uma Thread para um bloco específico de código, Usa-se a Classe `java.lang.Thread`.

quando fazemos o `“start();”` já estamos iniciando o processamento paralelo, e liberando o programa para executar qualquer outra thread. Então tenha a seguinte idéia assimilada: Se você deseja que o programa não “trave para o usuário” naquele determinado procedimento que pode demorar muito tempo, use Thread.

Caso você deseja que sua classe seja processada paralelamente, porém ela já estende de outra, você poderá optar por implementar o Runnable, que é a interface padrão para Thread. Por boas práticas, geralmente implementamos Runnable em vez de estender de Thread.

Utilize os recursos da programação concorrente com cuidado e só nos momentos que de fato precisar. Não saia de forma alguma colocando Runnable em todos os códigos pensando que seu programa ficará mais rápido, pelo contrário, um processo pode precisar de recursos de outro que ainda não está disponível e assim por diante, analise com cuidado a necessidade do uso deste poderoso recurso.

Um exemplo mais comum ainda de ser visto é a barra de progresso presente na maioria dos softwares. Enquanto está carregando um documento X é mostrada uma barra de progresso para o usuário saber que o software não travou.