



## Por Que não Paralelizar?

**Nome: Samuel Bozza de Moraes**

**Matricula: 202208565051**

**Inserir aqui o Campus: Estácio de Sa**

**Inserir aqui o nome da Disciplina: Por Que não Paralelizar? – Turma 2022.3 –  
Semestre Letivo: Mundo 3**

### Objetivo da Prática

Descreva nessa seção qual o objetivo da sua prática. Todos os Relatórios de Práticas deverão ser confeccionados em arquivo no formato PDF, com a **Logo da Universidade, nome do Campus, nome do Curso, nome da Disciplina, número da Turma, semestre letivo**. Além disso, o projeto deve ser armazenado em um repositório no GIT e o respectivo endereço deve constar na documentação e essa documentação deve estar no no GIT. O código deve estar versionado no GIT de forma organizada.

Lembre-se que a organização contará pontos.

Esse template é um modelo a ser seguido. O aluno pode optar por seguir outro modelo, **desde que atenda a todas as etapas disponíveis na Missão Prática**. O documento final deve estar em pdf.

### 1º Procedimento | Criando o Servidor e Cliente de Teste

Inserir neste campo, **de forma organizada**, todos os códigos do roteiro do 1º Procedimento da Atividade Prática, os resultados da execução do código e a Análise e Conclusão:

A. Como funcionam as classes Socket e ServerSocket?

R: são portas de entrada para processos de comunicação, isto é, eles permitem que dois processos destinates se comuniquem e, por consequência, geram determinado resultado

B. Qual a importância das portas para a conexão com servidores?

R: permitem que os computadores diferenciem facilmente entre diferentes tipos de tráfego

C. Para que servem as classes de entrada e saída `ObjectInputStream` e `ObjectOutputStream`, e por que os objetos transmitidos devem ser serializáveis?

R: São usadas para serialização e desserialização de objetos. A serialização converte objetos em uma sequência de bytes e a desserialização reconstrói objetos a partir dessa sequência. É usar o `Serializable` para garantir que a serialização e desserialização ocorram corretamente.

D. Por que, mesmo utilizando as classes de entidades JPA no cliente, foi possível garantir o isolamento do acesso ao banco de dados?

R: Ao usar entidades JPA no cliente, o isolamento do acesso ao banco de dados é assegurado pela gestão de transações, cache de primeiro nível, controle de contexto de persistência, estratégias de carregamento eficientes e delimitação apropriada do escopo da transação.

Observe que os tópicos acima seguem exatamente o que está na Atividade Prática exigida.

## 2º Procedimento | Servidor Completo e Cliente Assíncrono

Inserir neste campo, **de forma organizada**, todos os códigos do roteiro do 2º Procedimento da Atividade Prática, os resultados da execução do código e a Análise e Conclusão:

A. Como as Threads podem ser utilizadas para o tratamento assíncrono das respostas enviadas pelo servidor?

R: Threads podem ser utilizadas para tratar respostas assíncronas do servidor de diversas maneiras, como através de criação de threads individuais para cada requisição, utilizando pools de threads para reutilização eficiente.

B. Para que serve o método `invokeLater`, da classe `SwingUtilities`?

R: é usado para agendar a execução de uma tarefa na fila de eventos do Swing

C. Como os objetos são enviados e recebidos pelo Socket Java?

R: Em Java, para enviar e receber objetos por meio de sockets, utiliza-se a serialização e desserialização.

D. Compare a utilização de comportamento assíncrono ou síncrono nos clientes com Socket Java, ressaltando as características relacionadas ao bloqueio do processamento.

R: Comportamento síncrono em sockets Java envolve bloqueio de threads até a conclusão da operação, facilitando a programação sequencial, mas podendo levar à ociosidade com múltiplas threads bloqueadas.

Comportamento assíncrono permite que a thread principal continue execução enquanto espera, sendo mais complexo devido a callbacks, mas potencialmente mais eficiente em sistemas com muitas conexões simultâneas e evitando ociosidade.

Observe que os tópicos acima seguem exatamente o que está na Atividade Prática exigida.

## **Conclusão**

Elabore uma análise crítica da sua Missão Prática.

Usei o IntelliJ para fazer a prática e tive dificuldades para fazer a conexão do Cliente para o Servidor me levando a pensar que a IDE NetBeans seria mais apropriada para uso de sockets e threads.