



Douglas Justino Monteiro 202211506973

**Polo Centro – Olimpia-SP – Desenvolvimento full stack
Por que não paralelizar? – Número da Turma – 3º Semestre**

1º procedimento:

5 – Analise e conclusão

- a. **Socket:** esta classe é usada pelo servidor para declarar um objeto socket, que o servidor usa para enviar e receber dados do cliente.
Serversocket: esta classe é usada pelo servidor para declarar um objeto serversocket que o servidor precisa para executar as solicitações de conexão dos clientes.
- b. **O proposito das portas é para singularmente identificar aplicações e processos de um único computador e assim possibilita-los a compartilhar uma única conexão física com uma rede de comunicação de pacotes, como a internet.**
- c. **Elas são responsáveis por inserir e recuperar objetos serializados do stream. Uma vez serializado o objeto pode ser salvo em arquivos ou transmitido remotamente usando uma rede de computadores através de um fluxo de dados ou stream via HTTP, via socket, entre outros.**
- d. **Por causa do nível de isolamento.**

2º procedimento

5 – Analise e conclusão

- a. **Podem ser utilizadas de maneira que permite que o servidor realize o trabalho de forma assíncrona em threads paralelas.**
- b. **O que o método invokeLater faz é receber um evento de interface (encapsulando numa implementação de Runnable) e enfileirar-lo, para que seja despachado pela thread de eventos de interface assim que houver oportunidade.**
- c. **Um exemplo de um cliente que solicita arquivos ao servidores. O cliente deve informar o caminho do arquivo no servidor, ao receber o pedido o servidor enviara o arquivo ao cliente. A aplicação solicita um arquivo presente em um diretório do servidor, e ao receber o arquivo, exibe um JLabel o conteúdo do arquivo.**

Para cada solicitação do cliente será criada uma nova thread para gerenciar a troca de mensagens.

d.Analogia: Na Assíncrona cada bola tem um número de sequência, que permite que seja colocada na sua posição. Isto permite que as bolas sejam enviadas e recebidas por qualquer ordem.

Na síncrona as bolas têm que ser enviadas e recebidas de forma sincronizada, mantendo uma ordem bem definida, a primeira bola a ser enviada, tem que ser a primeira a ser recebida e assim sucessivamente.

O bloqueio de processamento acontece na forma síncrona, pois enquanto uma thread está sendo executada as outras permanecem bloqueadas.