



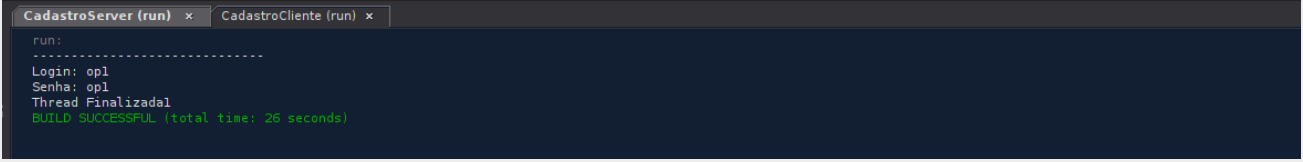
Servidores e clientes baseados em Socket, com uso de Threads tanto no lado cliente quanto no lado servidor, acessando o banco de dados via JPA.

Codigo Da Aula

<https://github.com/adventureandre/Server-Cliente-Java-JPA-Socket>

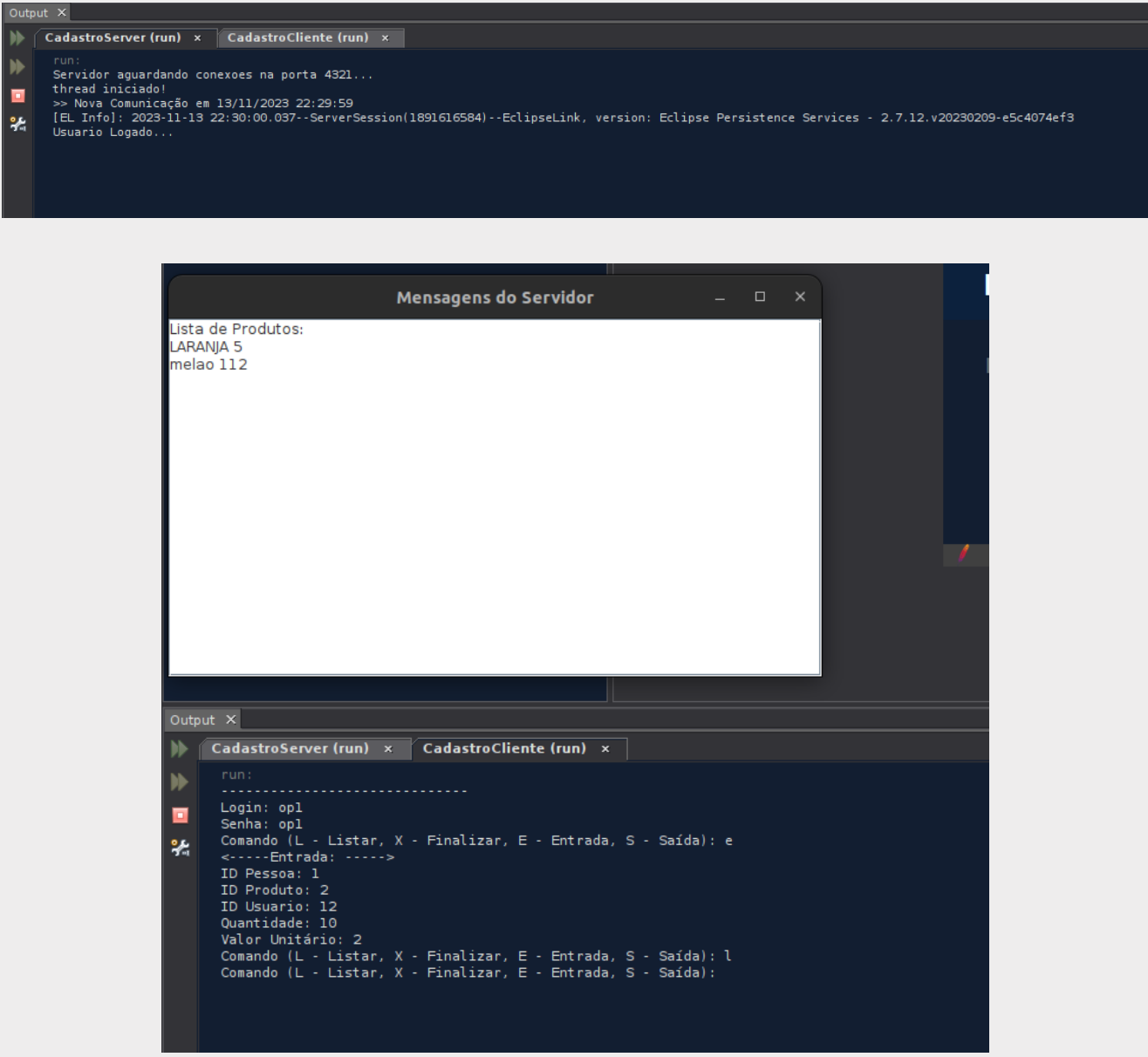
1º Procedimento | Criando o Servidor e Cliente de Teste

Resultado 1º Procedimento



2º Procedimento | Servidor Completo e Cliente Assíncrono

Resultado 2º Procedimento



Análise e Conclusão 1º Procedimento

O primeiro procedimento abordou a criação de um servidor e cliente síncrono utilizando Java Sockets. O servidor autentica usuários, responde a comandos como "L" para listar produtos e interage com um banco de dados via JPA. A implementação eficiente de Threads no servidor permite lidar com múltiplas conexões simultaneamente, contribuindo para a escalabilidade do sistema. O cliente realiza operações básicas como autenticação, envio de comandos e recebimento de respostas do servidor, proporcionando uma base sólida para procedimentos subsequentes.

Análise e Conclusão 2º Procedimento

O segundo procedimento expandiu o servidor para incluir funcionalidades adicionais, permitindo entrada e saída de produtos. Introduziu também um cliente assíncrono que utiliza Threads para possibilitar a execução de comandos de forma assíncrona, evitando bloqueios durante a interação. A inclusão de uma janela de mensagens no cliente e a implementação de comandos mais elaborados ampliam a aplicabilidade e a usabilidade do sistema. Ambos os procedimentos demonstram o uso prático de Threads e Sockets para implementar comunicação cliente-servidor eficiente em Java.