
Exchanger em Java - Mudança de Dados entre Tarefas Concorrentes



STATUS

DESENVOLVIDA

Objetivo do Projeto



Este projeto demonstra o uso da classe Exchanger para coordenar a troca de dados entre threads em Java. O sistema consiste em duas classes principais:

Producer: Produz mensagens e as coloca em um buffer. Consumer: Consome as mensagens do buffer trocado com o Producer. A comunicação entre as threads ocorre através do Exchanger, garantindo que a troca de dados aconteça de forma sincronizada.

Estrutura do Código

O projeto é composto pelos seguintes arquivos:

Producer.java: Implementa a lógica de produção de eventos. Consumer.java: Implementa a lógica de consumo de eventos. Main.java: Configura os buffers e inicia as threads para demonstrar a troca de dados.

Funcionamento

O Producer cria 10 mensagens por ciclo e as armazena em um buffer. O Consumer espera pelo buffer preenchido e processa as mensagens. As threads trocam os buffers através do Exchanger. O processo se repete por 10 ciclos.



Cenários teste



Cenário 1: Execução Normal

O Producer produz 10 mensagens por ciclo e as troca com o Consumer. O Consumer recebe e consome as mensagens corretamente. O programa funciona sem condições de corrida.

=====

Cenário 2: Buffer Vazio

O Consumer tenta acessar um buffer vazio antes da troca ocorrer. O Consumer aguarda até que o Producer esteja pronto. O sincronismo é mantido corretamente.

=====

Cenário 3: Múltiplos Ciclos

O Producer e o Consumer completam 10 ciclos de troca. Todas as mensagens são trocadas e processadas corretamente.



Conclusão:



O uso do Exchanger se mostrou eficaz para sincronizar threads na troca de dados. Os testes realizados confirmam que as mensagens são trocadas corretamente, sem perda ou duplicidade. Com algumas melhorias, o sistema pode se tornar ainda mais robusto e eficiente.



Desenvolvedor



Marcio Fonseca



=====