
Sistema de Controle de Preços com ReadWriteLock - Sincronização de Acesso a Dados com Read/Write Locks



STATUS

DESENVOLVIDA

Objetivo do Projeto



Este documento descreve a implementação e os testes realizados em um sistema onde múltiplas threads leem e escrevem em um recurso compartilhado chamado PricesInfo. O sistema utiliza ReadWriteLock para garantir a consistência dos dados e evitar condições de corrida.

Estrutura do Código

Classe PricesInfo

Contém dois preços (price1 e price2). Utiliza ReadWriteLock para controle de concorrência. Possui os métodos getPrice1() e getPrice2(), protegidos por bloqueios de leitura (readLock). Possui o método setPrices(), protegido por bloqueio de escrita (writeLock).

Classe Reader

Implementa a interface Runnable. Lê os preços 10 vezes, com um pequeno intervalo entre cada leitura.

Classe Writer

Implementa a interface Runnable. Modifica os preços 3 vezes, com um pequeno intervalo entre cada escrita.

Classe Main

Cria uma instância de PricesInfo. Inicia 5 threads leitoras e 1 thread escritora. Gerencia a execução concorrente dos acessos aos preços.



Cenários teste



Cenário 1: Múltiplas Leituras Simultâneas

Objetivo: Verificar se múltiplas threads podem ler os preços simultaneamente.

Resultado: As threads leitoras conseguem acessar os preços ao mesmo tempo, sem bloqueio.

=====

Cenário 2: Escrita com Exclusividade

Objetivo: Verificar se a escrita bloqueia as leituras.

Resultado: Durante a escrita, as leituras são bloqueadas.

=====

Cenário 3: Concorrência entre Leituras e Escrita

Objetivo: Testar a interação entre threads leitoras e escritora.

Resultado: Leituras são bloqueadas durante a escrita, garantindo a consistência dos dados.

Conclusão:



O uso de ReadWriteLock permite um acesso eficiente aos preços, permitindo múltiplas leituras simultâneas e garantindo exclusividade durante a escrita. Os testes realizados confirmam que o mecanismo de bloqueio está funcionando corretamente, evitando condições de corrida e garantindo a consistência dos dados no sistema.

Desenvolvedor



Marcio Fonseca



=====