



Java

Synchronized





O que é o método `synchronized` em Java e qual é o seu propósito principal?

O método `synchronized` é utilizado para controlar o acesso concorrente a blocos de código ou métodos. Ele garante que apenas uma thread por vez execute o código sincronizado, prevenindo condições de corrida e garantindo consistência nos dados compartilhados.



Como o bloqueio intrínseco é relacionado ao método `synchronized`?

O método `synchronized` usa o conceito de bloqueio intrínseco para garantir a exclusão mútua. Cada objeto Java possui um monitor associado a ele, e apenas uma thread por vez pode possuir o monitor.

Quando uma thread entra em um bloco `synchronized` ou método, ela adquire o bloqueio intrínseco do objeto, impedindo que outras threads executem código sincronizado nesse objeto até que o bloqueio seja liberado.



Qual é a diferença entre o uso de um bloco `synchronized` e a sincronização de um método inteiro?

Um bloco `synchronized` permite que você especifique uma seção crítica do código que deve ser protegida contra acesso concorrente. Por outro lado, a sincronização de um método inteiro implica que todo o método é considerado crítico.

A escolha entre os dois depende dos requisitos específicos de sincronização e do escopo desejado para evitar condições de corrida.



Quais são os possíveis problemas associados ao uso excessivo de métodos `synchronized` em uma aplicação Java?

: O uso excessivo pode levar a problemas de desempenho, pois pode resultar em bloqueios desnecessários e maior contenção. Além disso, é importante evitar bloqueios muito longos, pois isso pode aumentar o risco de deadlock.

A sincronização excessiva também pode limitar a escalabilidade de uma aplicação em um ambiente concorrente.



luisfabriciodellamas