

Lista de exercícios sobre RMI e RPC

1. No contexto de RMI, explique como é realizada a serialização e desserialização de objetos. Quais são os desafios associados à serialização de objetos complexos e como podem ser superados?
2. Em um ambiente RMI, como você lidaria com a garantia de consistência transacional em operações que envolvem chamadas de métodos remotos? Quais são os padrões e abordagens para implementar transações distribuídas?
3. O mecanismo de stubs e skeletons é uma parte essencial do RMI. Explique a função deles e como eles facilitam a chamada de métodos remotos.
4. Em RMI, o registro de objetos é fundamental para permitir que os clientes encontrem objetos remotos. Discuta os desafios de gerenciar esse registro em um ambiente distribuído e como a descoberta de serviços pode ser melhorada.
5. Ao projetar uma aplicação RMI, como você garantiria a segurança e a autenticação das chamadas de métodos remotos? Quais são as práticas recomendadas para prevenir ataques como man-in-the-middle e spoofing?
6. Considerando que RMI é fortemente associado à plataforma Java, como você estenderia a interoperabilidade entre sistemas baseados em diferentes tecnologias, como Java e .NET? Quais são as estratégias para lidar com esse tipo de heterogeneidade?
7. Explique a diferença entre RMI síncrono e assíncrono. Quando optar por um ou outro modelo de chamada de método remoto e como cada um afeta a arquitetura e o desempenho do sistema?
8. Em uma aplicação RMI distribuída de larga escala, como você otimizaria o desempenho e minimizaria a latência nas comunicações entre os componentes? Quais são os trade-offs entre desempenho e complexidade do código?
9. RMI pode ser afetado por problemas como a "marshalling hell", onde a evolução de classes pode causar incompatibilidades na serialização de objetos. Como você gerenciaría esse problema ao longo do ciclo de vida de uma aplicação RMI?
10. Considerando os avanços tecnológicos e as mudanças nas arquiteturas de sistemas distribuídos, discuta como RMI se encaixa em cenários modernos, como arquiteturas de microsserviços e aplicações em nuvem.
11. Explique como RMI fornece uma abstração transparente para os desenvolvedores ao lidar com a comunicação entre objetos distribuídos. Como essa abstração é alcançada e qual é a vantagem dela?

12. Quais são as etapas necessárias para expor um objeto Java como um objeto remoto utilizando RMI? Como o registro de objetos remotos é facilitado por essa abstração?
13. Com RMI, os desenvolvedores podem invocar métodos de objetos remotos como se estivessem chamando métodos locais. Como essa abstração é possível e quais são os componentes-chave envolvidos nesse processo?
14. Uma das facilidades de RMI é a transmissão transparente de parâmetros e resultados entre objetos remotos. Explique como RMI lida com a serialização e desserialização automática de dados complexos.
15. Uma das facilidades de RMI é a transmissão transparente de parâmetros e resultados entre objetos remotos. Explique como RMI lida com a serialização e desserialização automática de dados complexos.
16. Uma das facilidades de RMI é a transmissão transparente de parâmetros e resultados entre objetos remotos. Explique como RMI lida com a serialização e desserialização automática de dados complexos.