

Este código implementa uma aplicação distribuída usando Java RMI (Remote Method Invocation), permitindo que um cliente invoque métodos remotos (neste caso, operações matemáticas) que estão hospedados em um servidor.

Estrutura do Código

1. **InterfaceServidorMat.java:**

- Define a interface remota que declara métodos que o servidor disponibilizará para o cliente.
- Esta interface estende `Remote`, o que significa que seus métodos podem ser chamados remotamente.
- Os métodos disponíveis são `soma`, `subtrai`, `multiplica`, e `divide`, todos lançando a exceção `RemoteException` para tratar problemas de comunicação entre o cliente e o servidor.

2. **ServidorMat.java:**

- Implementa a interface `InterfaceServidorMat` e define a lógica dos métodos remotos.
- A classe `ServidorMat` estende `UnicastRemoteObject`, o que permite que as instâncias sejam exportadas para receber chamadas remotas.
- Cada método realiza uma operação matemática básica.
- Quando o servidor é inicializado, ele exibe uma mensagem para indicar que está ativo e pronto para responder aos clientes.

3. **ArrancaServidor.java:**

- Contém o método `main` que registra o objeto `ServidorMat` no RMI Registry.
- `Naming.rebind("ServidorMat_1", new ServidorMat());` registra o objeto com o nome `"ServidorMat_1"`, tornando-o acessível para chamadas remotas usando esse identificador.
- Se ocorrer um erro ao inicializar o servidor, uma mensagem de erro é exibida.

4. **Cliente.java:**

- O cliente tenta localizar o serviço remoto `ServidorMat_1` registrado no RMI Registry.
- Se o servidor estiver ativo e acessível, o cliente pode usar a interface `InterfaceServidorMat` para realizar operações remotas.
- Os métodos `area` e `perimetro` realizam cálculos baseados nas operações do servidor:
 - `area(a, b)`: calcula a área multiplicando `a` por `b`.
 - `perimetro(a, b)`: calcula o perímetro, somando `a` e `b` e multiplicando o resultado por 2.
- No `main`, o cliente instancia `Cliente` e exibe os resultados da `area` e `perimetro`.

Passos para Rodar o Projeto no IntelliJ IDEA

1. **Configuração do RMI Registry:**

- Execute o RMI Registry a partir do terminal (no mesmo diretório onde estão as classes compiladas):

start rmiregistry

- Deixe essa janela do terminal aberta para o registry continuar rodando.
- 2. **Compilação e Execução no IntelliJ:**
 - Certifique-se de que o projeto está configurado corretamente no IntelliJ e que o RMI está habilitado.
 - **Compilação:**
 - Compile todas as classes.
 - **Execução do Servidor:**
 - Primeiro, execute `ArrancaServidor`. Isso registra o servidor no registry RMI.
 - Verifique se a saída no console mostra "Arrancando Servidor..." e "Novo Servidor Instanciado...".
 - **Execução do Cliente:**
 - Execute `Cliente`. O cliente tentará conectar ao servidor RMI, realizar as operações e imprimir os resultados de área e perímetro.
 - A saída deve mostrar os resultados dos cálculos remotos e indicar que a comunicação entre cliente e servidor foi bem-sucedida.

Esses passos permitirão que você configure e rode o projeto RMI no IntelliJ, possibilitando que o cliente acesse e use os métodos remotos expostos pelo servidor.