

Sistemas Distribuídos 2021/2022

Ficha Prática Nº 7 - Java RMI (cont.)

Implementar um programa cliente-servidor para realizar leilões, utilizando Java RMI. O programa deve utilizar uma classe básica chamada Item, conforme a definição abaixo, a ser completada:

```
public class Item{
    private int id;
    private String descricao;
    private Calendar dataHoraFecho;
    private float precoMinimo;
    private float precoAtual;
    private int numeroOfertas;
    private boolean terminado;

    public Item(String descrição, Calendar dataHoraFecho,
                float precoMinimo){...}

    //-- getters and setters
    ...
};
```

Observação:

- O atributo “id” é atribuído pelo servidor, de forma sequencial, a iniciar por 1.
- O servidor é responsável por alterar o atributo “terminado” antes de retornar informações de um item aos clientes, ou seja, quem determina se o leilão de um item foi ou não terminado é o servidor, com base na sua data/hora.

A interface remota do servidor (chamar de InterfaceLeilao) deve disponibilizar os seguintes métodos aos clientes:

```
Item      obtemItem(int id);
Item[ ]   obtemTodosItens();
boolean   adicionaItem(Item novo);
boolean   removeItem(int id);
boolean   oferece(int id, float valor);
```

As variáveis booleanas que retornam de alguns dos métodos descritos servem para indicar condições de erro. Da mesma forma, o retorno de um objeto nulo (null) serve para indicar que não foi encontrado o item pelo método obtemItem. Apenas uma única lista de itens de leilão deve ser mantida no servidor, sendo comum a todos os clientes. Os programas devem utilizar as classes LocateRegistry e Registry.

O programa cliente deve implementar a seguinte funcionalidade:

- Apresentar um resumo de todos os leilões existentes (id, descrição e se está terminado).
- Apresentar todos os dados de um item – deve apresentar mensagem de erro se o item não existe.
- Adicionar um item. Apenas é necessário obter do utilizador os parâmetros definidos no construtor da classe Item. Deve apresentar mensagens de erro se contiver campos nulos ou inválidos.
- Remover um item – deve apresentar mensagem de erro se o item não existe.
- Fazer uma oferta a um item – deve apresentar mensagem de erro se o item não existe, ou o valor ofertado seja inválido, ou o leilão já tenha terminado.
- Sair do programa cliente.

Pergunta e explicações:

A exclusão mútua no acesso aos métodos de um objeto remoto é garantida pelo Java RMI? Explique e **demonstre** a sua resposta. Incluir *screenshots* dos testes realizados (sobre exclusão mútua). Caso não seja garantida exclusão mútua, implementar os mecanismos necessários para evitar *race-conditions* e explicar os mecanismos utilizados.

Submeter os 2 projetos Netbeans (servidor e cliente) e as respostas em pdf no InforEstudante.