

Sistemas Distribuídos

Trabalho 2

João Vitor Narciso, João Pedro Medeiros e Matheus

Interfaces

Client

```
ClientInterface.java
1  import java.rmi.Remote;
2  import java.rmi.RemoteException;
3
4  public interface ClientInterface extends Remote {}
```

Catalog

```
CatalogInterface.java
1  import java.rmi.Remote;
2  import java.rmi.RemoteException;
3
4  public interface CatalogInterface extends Remote {
5      public String visualizar(int id) throws RemoteException;
6
7      public String registrar(int id, String name) throws RemoteException;
8
9      public String remover(int id) throws RemoteException;
10
11     public String vender(int id) throws RemoteException;
12 }
13
```

Billing

```
BillingInterface.java
1  import java.rmi.Remote;      You, 32 minutes ago • ex nihilo nihil fit
2  import java.rmi.RemoteException;
3
4  public interface BillingInterface extends Remote {
5      public String vender(int id, int valor, String metodoPagamento) throws RemoteException;
6
7      public String emitirNota(int id, int valor, int cnpj) throws RemoteException;
8
9      public String cancelarVenda(int id) throws RemoteException;
10 }
11
```

Warehouse

```
WarehouseInterface.java > ...
You, 33 minutes ago | 1 author (You)
1  import java.rmi.Remote;
2  import java.rmi.RemoteException;
3
4  You, 33 minutes ago | 1 author (You)
5  public interface WarehouseInterface extends Remote {
6      public String despachar(int id, String endereco) throws RemoteException;
7
8      public String separar(int id) throws RemoteException;
9
10     public String extraviar(int id) throws RemoteException;
11 }
```

Conexão no Client

Primeiro salvamos os argumentos em variáveis para facilitar a organização e conexão dos serviços.

```
String clientIp = args[0];
String catalogIp = args[1];
String billingIp = args[2];
String warehouseIp = args[3];
int qtdVendas = Integer.parseInt(args[4]);
```

Depois montamos as strings de conexão:

```
String connectCatalog = "rmi://" + catalogIp + rmiCatalog;
String connectBilling = "rmi://" + billingIp + rmiBilling;
String connectWarehouse = "rmi://" + warehouseIp + rmiWarehouse;
```

Métodos

Por ser uma simulação, os métodos invocam `System.out.println` para simular as operações. Primeiro criamos um array de métodos de pagamento, um array de endereços e um array de produtos para, através de um índice aleatório fornecer como argumento para os métodos dos serviços:

```
String [] formaPagamento = {"Crédito", "Débito", "Dinheiro"};
int randomPagamento;
String [] endereco = {"casa 1", "casa 2", "casa 3", "casa 4" };
int randomEndereco;
String [] produtos = {"Meia", "Sapato", "Livro", "Notebook", "Mouse"};
int randomProduto;
```

Aleatoriedade

Para chamar métodos de forma aleatória e realizar simulações diferentes criamos um valor aleatório, em um intervalo de 4 números) para ser utilizado como case em um Switch, também geramos um preço aleatório no intervalo com o intervalo 5000:

Dentro de cada case existe um loop (for) para iterar e utilizar o índice como id do produto através de iterações aleatórias no intervalo de 6 números.

```

for (int i = 0; i < qtdVendas; i++) {
    int result = new Random().nextInt(4);
    int preco;
    int randomFor = new Random().nextInt(15);

    switch (result) {
        case 0:
            for (int j = 0; j < randomFor; j++) {
                randomPagamento = new Random().nextInt(3);
                randomProduto = new Random().nextInt(5);
                randomEndereco = new Random().nextInt(4);
                preco = new Random().nextInt(5000);
                System.out.println(catalog.visualizar(j));
                System.out.println(warehouse.despachar(j, endereco[randomEndereco]));
                System.out.println(billing.vender(j, preco, formaPagamento[randomPagamento]));
                System.out.println(catalog.remover(j));
            }
            break;

        case 1:
            for (int j = 0; j < new Random().nextInt(15); j++) {
                randomPagamento = new Random().nextInt(3);
                System.out.println(catalog.visualizar(j));
                System.out.println(warehouse.separar(j));
                System.out.println(billing.vender(j, new Random().nextInt(5000), formaPagamento[randomPagamento]));
                System.out.println(billing.cancelarVenda(j));
            }
            break;

        case 2:
            for (int j = 0; j < new Random().nextInt(15); j++) {
                randomPagamento = new Random().nextInt(3);
                randomProduto = new Random().nextInt(5);
                randomEndereco = new Random().nextInt(4);
                System.out.println(catalog.registrar(j, produtos[randomProduto]));
                System.out.println(catalog.visualizar(j));
                System.out.println(billing.vender(j, new Random().nextInt(5000), formaPagamento[randomPagamento]));
                System.out.println(warehouse.separar(j));
                System.out.println(warehouse.despachar(j, endereco[randomEndereco]));
            }
            break;

        default:
            for (int j = 0; j < new Random().nextInt(15); j++) {
                randomPagamento = new Random().nextInt(3);
            }
    }
}

```

Serviços rodando:

Compilando e rodando com o ip

```

jn@MacBook-Air-de-Administrador-3 T2 - Java RMI % javac *
jn@MacBook-Air-de-Administrador-3 T2 - Java RMI % java Catalog 192.168.5.25

java RMI registry created.
Catalog is ready.

```

```
jn@MacBook-Air-de-Administrador-3 T2 - Java RMI % java Billing 192.168.5.25
java RMI registry already exists.
Billing is ready.
```

```
jn@MacBook-Air-de-Administrador-3 T2 - Java RMI % java Warehouse 192.168.5.25
java RMI registry created.
Warehouse is ready.
```

Cada serviço retorna uma String que é exibida no Cliente, mas também lança uma `System.out.println` para garantir a exibição de logs específicos em cada serviço.

Cliente

Iniciando cliente:

```
$ java <client ip> <catalog ip> <billing ip> <warehouse ip> <numero vendas>
```

```
jn@MacBook-Air-de-Administrador-3 T2 - Java RMI % java Client 192.168.5.25 192.168.5.25 192.168.5.25 192.168.5.25 1
java RMI registry created.
Client is ready.
Connecting to catalog at : rmi://192.168.5.25:52370/Catalog
Catalog is ready.
Connecting to catalog at : rmi://192.168.5.25:52371/Billing
Billing is ready.
Connecting to catalog at : rmi://192.168.5.25:52372/Warehouse
Warehouse is ready.
Visualizando produtos...
```

Exibindo todos os logs de todos os serviços.

```
Visualizando produto 0
0 produto0 foi despachado para casa 4
0 produto0foi vendido por 3007via Débito
Produto com o id 0 foi removido.
Visualizando produto 1
0 produto1 foi despachado para casa 2
0 produto1foi vendido por 2382via Débito
Produto com o id 1 foi removido.
Visualizando produto 2
0 produto2 foi despachado para casa 1
0 produto2foi vendido por 3829via Dinheiro
Produto com o id 2 foi removido.
Visualizando produto 3
0 produto3 foi despachado para casa 2
0 produto3foi vendido por 4059via Dinheiro
Produto com o id 3 foi removido.
Visualizando produto 4
0 produto4 foi despachado para casa 4
0 produto4foi vendido por 3753via Dinheiro
Produto com o id 4 foi removido.
Visualizando produto 5
0 produto5 foi despachado para casa 4
0 produto5foi vendido por 2347via Dinheiro
Produto com o id 5 foi removido.
Visualizando produto 6
0 produto6 foi despachado para casa 1
0 produto6foi vendido por 1875via Débito
Produto com o id 6 foi removido.
Visualizando produto 7
0 produto7 foi despachado para casa 4
0 produto7foi vendido por 139via Crédito
Produto com o id 7 foi removido.
Visualizando produto 8
0 produto8 foi despachado para casa 3
0 produto8foi vendido por 3360via Dinheiro
Produto com o id 8 foi removido.
Visualizando produto 9
0 produto9 foi despachado para casa 4
0 produto9foi vendido por 2275via Débito
Produto com o id 9 foi removido.
Visualizando produto 10
0 produto10 foi despachado para casa 1
0 produto10foi vendido por 1641via Débito
Produto com o id 10 foi removido.
Visualizando produto 11
0 produto11 foi despachado para casa 2
0 produto11foi vendido por 2305via Dinheiro
Produto com o id 11 foi removido.
Visualizando produto 12
0 produto12 foi despachado para casa 3
0 produto12foi vendido por 2275via Débito
Produto com o id 12 foi removido.
```

Serviços exibindo seus respectivos logs

- Catalog

```

jn@MacBook-Air-de-Administrador-3 T2 - Java RMI % java Catalog 192.168.5.25
java RMI registry created.
Catalog is ready.
Visualizando produto 0
Visualizando produto 0
Produto com o id 0 foi removido.
Visualizando produto 1
Produto com o id 1 foi removido.
Visualizando produto 2
Produto com o id 2 foi removido.
Visualizando produto 3
Produto com o id 3 foi removido.
Visualizando produto 4
Produto com o id 4 foi removido.
Visualizando produto 5
Produto com o id 5 foi removido.
Visualizando produto 6
Produto com o id 6 foi removido.
Visualizando produto 7
Produto com o id 7 foi removido.
Visualizando produto 8
Produto com o id 8 foi removido.
Visualizando produto 9
Produto com o id 9 foi removido.
Visualizando produto 10
Produto com o id 10 foi removido.
Visualizando produto 11
Produto com o id 11 foi removido.
Visualizando produto 12
Produto com o id 12 foi removido.

```

- Billing

```

jn@MacBook-Air-de-Administrador-3 T2 - Java RMI % java Billing 192.168.5.25
java RMI registry created.
Billing is ready.
0 produto0foi vendido por 2011via Crédito
Produto com o id 0 foi cancelado.
0 produto0foi vendido por 3007via Débito
0 produto1foi vendido por 2382via Débito
0 produto2foi vendido por 3829via Dinheiro
0 produto3foi vendido por 4059via Dinheiro
0 produto4foi vendido por 3753via Dinheiro
0 produto5foi vendido por 2347via Dinheiro
0 produto6foi vendido por 1875via Débito
0 produto7foi vendido por 139via Crédito
0 produto8foi vendido por 3360via Dinheiro
0 produto9foi vendido por 2275via Débito
0 produto10foi vendido por 1641via Débito
0 produto11foi vendido por 2305via Dinheiro
0 produto12foi vendido por 2275via Débito

```

- Warehouse


```
jn@MacBook-Air-de-Administrador-3 T2 - Java RMI % java Warehouse 192.168.5.25
java RMI registry created.
Warehouse is ready.
Produto id 0 foi separado...
0 produto0 foi despachado para casa 4
0 produto1 foi despachado para casa 2
0 produto2 foi despachado para casa 1
0 produto3 foi despachado para casa 2
0 produto4 foi despachado para casa 4
0 produto5 foi despachado para casa 4
0 produto6 foi despachado para casa 1
0 produto7 foi despachado para casa 4
0 produto8 foi despachado para casa 3
0 produto9 foi despachado para casa 4
0 produto10 foi despachado para casa 1
0 produto11 foi despachado para casa 2
0 produto12 foi despachado para casa 3
```

Link pro Github: <https://github.com/jotanarciso/javarmi>