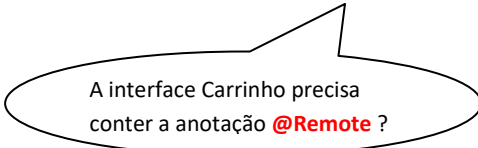


Exercício 4 – Parte 2

Criar uma aplicação para consumir o EJB3 CarrinhoBean recém criado. Favor anotar o tempo gasto para implementar esse exercício.

- 1 – criar um projeto Java no Eclipse com o nome de **2.1-ejb3-cliente** (File>New>Java Project).
- 2 – criar os pacotes **br.com.fiap.ejb3** e **br.com.fiap.ejb3.cliente**.
- 3 – copiar a classe **Item** e a interface **Carrinho** do projeto **2.1-ejb3** para o pacote **br.com.fiap.ejb3**.



A interface Carrinho precisa conter a anotação **@Remote** ?

- 4 – criar e executar a classe abaixo que realizará a chamada ao EJB3 recém criado.

```
package br.com.fiap.ejb3.cliente;

import java.util.Properties;

import javax.naming.Context;
import javax.naming.InitialContext;
import javax.naming.NamingException;

import br.com.fiap.ejb3.Carrinho;
import br.com.fiap.ejb3.Item;

public class CarrinhoCliente {

    public static void main(String[] args) throws NamingException {

        //propriedades necessarias para acessar o JNDI
        Properties props = new Properties();
        props.setProperty("java.naming.factory.initial",
            "org.jnp.interfaces.NamingContextFactory");
        props.setProperty("java.naming.provider.url", "localhost:1099");
        //props.setProperty("java.naming.provider.url", "10.1.21.41:1099");

        Context context = new InitialContext(props);

        Carrinho carrinho = (Carrinho)
            context.lookup("CarrinhoBean/remote");

        Item item1 = new Item();
        item1.setNome("Doril");
        item1.setPreco(3.0);

        carrinho.cadastrarItem(item1);
        System.out.println("Quantidade de produtos no carrinho: " +
            carrinho.obterQuantidadeItens());

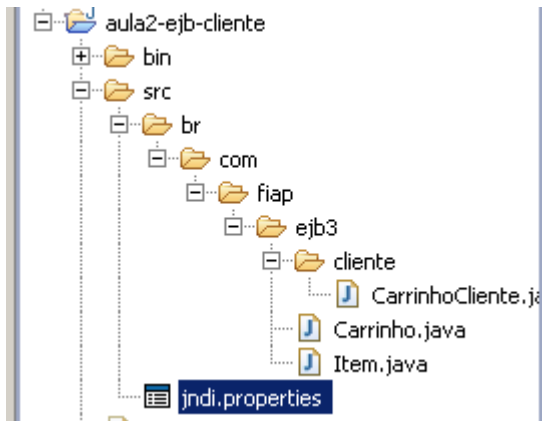
    }

}
```

Para executar o cliente do EJB também é necessário adicionar as bibliotecas do provider em uso (JBoss). Para tal adicione as bibliotecas do JBoss nas propriedades do projeto (**Java Build Path>Libraries>Add Library>Server Runtime>JBoss AS 6**)

4.1 – A utilização da classe **Properties** acima pode ser substituída pela adição de um arquivo de configuração chamado **jndi.properties**.

Este arquivo deve ficar na raiz do projeto e é lido automaticamente pela classe **InitialContext**.



Para testar crie o arquivo texto **jndi.properties** com o conteúdo abaixo

```
java.naming.factory.initial=org.jnp.interfaces.NamingContextFactory
java.naming.provider.url=localhost:1099
```

e comente as linhas de código a seguir na classe **CarrinhoCliente**.

```
props.setProperty("java.naming.factory.initial",
"org.jnp.interfaces.NamingContextFactory");
props.setProperty("java.naming.provider.url", "localhost:1099");
```

Execute novamente a classe **CarrinhoCliente** e veja o resultado. Em seguida faça um novo teste alterando o caminho "localhost" para o endereço de outra máquina. Esta é uma maneira de parametrizar o caminho do servidor evitando que este endereço fique fixo (hard coded) no código.

5 – Registrar o tempo gasto até aqui.