

TDM03 d'Informatique Répartie

Corba en Java

ASI4 - INSA Rouen
CORRECTION

Calculatrice distribuée sur entiers positifs à l'aide de Corba, en Java

Le but de ce TP est de réaliser une calculatrice distribuée sur entiers positifs à l'aide de Corba. Pour cela, vous respecterez les contraintes suivantes :

- Votre calculatrice supportera 4 types d'opération : additions, soustractions, multiplications et divisions.
- Elle ne travaillera que sur les entiers positifs.
- Le serveur proposera donc 4 services distribués.
- Le serveur et le client sont en Java. Vous n'utiliserez pas de service de nommage.
- Une exception sera levée en cas de division par 0 (exception existante), si un des opérandes ou si le résultat est négatif (exception à créer).

NB : il s'agit d'Informatique répartie donc pensez à tester votre programme avec serveur et client sur des machines différentes !

Correction

CalculatriceEntiers.idl

```
module calculatriceSurEntiers {  
    interface Calculatrice {  
        exception EntierNegatif {  
            string raison;  
        };  
        long additionner(in long e1, in long e2) raises (EntierNegatif);  
        long soustraire(in long e1, in long e2) raises (EntierNegatif);  
        long multiplier(in long e1, in long e2) raises (EntierNegatif);  
        long diviser(in long e1, in long e2) raises (EntierNegatif);  
    };  
};
```

calculatriceSurEntiers/CalculatriceServant.java

```
import calculatriceSurEntiers.*;  
import calculatriceSurEntiers.CalculatricePackage.*;  
  
class CalculatriceServant extends CalculatricePOA {  
    public int additionner(int e1, int e2) throws EntierNegatif {  
        if(e1<0)  
            throw new EntierNegatif("Negative operande: " + e1);  
        if(e2<0)  
            throw new EntierNegatif("Negative operande: " + e2);  
        return e1+e2;  
    }  
  
    public int soustraire(int e1, int e2) throws EntierNegatif {  
        int resultat;  
        if(e1<0)  
            throw new EntierNegatif("Negative operande: " + e1);  
        if(e2<0)  
            throw new EntierNegatif("Negative operande: " + e2);  
        resultat = e1-e2;  
        if(resultat<0)  
            throw new EntierNegatif("Negative result: " + resultat);  
        return resultat;  
    }  
  
    public int multiplier(int e1, int e2) throws EntierNegatif {  
        if(e1<0)  
            throw new EntierNegatif("Negative operande: " + e1);  
        if(e2<0)  
            throw new EntierNegatif("Negative operande: " + e2);  
        return e1*e2;  
    }  
  
    public int diviser(int e1, int e2) throws EntierNegatif {  
        if(e1<0)
```

```

        throw new EntierNegatif("Negative operande: " + e1);
    if(e2<0)
        throw new EntierNegatif("Negative operande: " + e2);
    return e1/e2;
    }
}

calculatriceSurEntiers/Serveur.java

import calculatriceSurEntiers.*;
import calculatriceSurEntiers.CalculatricePackage.*;

import org.omg.CORBA.*;
import org.omg.PortableServer.*;
import org.omg.PortableServer.POA;

class Serveur {
    public static void main(String args[]) {
        try {
            String orbArguments[]=new String[0];
            ORB orb=ORB.init(orbArguments,null);
            POA rootpoa = POAHelper.narrow(orb.resolve_initial_references("RootPOA"));
            rootpoa.the_POAManager().activate();
            CalculatriceServant servant = new CalculatriceServant();
            Calculatrice calculatriceCorbaObject = CalculatriceHelper.narrow(rootpoa.servant_to_reference(servant));
            System.out.println(orb.object_to_string(calculatriceCorbaObject));
            orb.run();
        }
        catch (Exception e) {
            System.err.println("Error : "+e);
        }
    }
}

```

```

calculatriceSurEntiers/Client.java

import calculatriceSurEntiers.*;
import calculatriceSurEntiers.CalculatricePackage.*;

import org.omg.CosNaming.*;
import org.omg.CORBA.*;
import java.lang.Integer;

public class Client {
    public static void main(String args[]) {
        try {
            String orbArguments[] = new String[0];
            ORB orb = ORB.init(orbArguments, null);
            org.omg.CORBA.Object corbaObject = orb.string_to_object(args[0]);
            Calculatrice calculatriceCorbaObject = CalculatriceHelper.narrow(corbaObject);
            int operande1 = Integer.parseInt(args[args.length-3]),
                operande2 = Integer.parseInt(args[args.length-1]),
                resultat=-1;

            if(args[args.length-2].equals("+"))
                resultat = calculatriceCorbaObject.additionner(operande1, operande2);
            if(args[args.length-2].equals("-"))
                resultat = calculatriceCorbaObject.soustraire(operande1, operande2);
            if(args[args.length-2].equals("x"))
                resultat = calculatriceCorbaObject.multiplier(operande1, operande2);
            if(args[args.length-2].equals("/"))
                resultat = calculatriceCorbaObject.diviser(operande1, operande2);

            System.out.println(operande1 + args[args.length-2] + operande2 + " = " + resultat);
        }
        catch (Exception e) {
            System.err.println("Error : "+e);
        }
    }
}

```

Remarques

- Vous aurez alors accès à la correction du TDM durant une semaine à compter de la fin du TDM.
- **Déposez un compte-rendu de 2 pages** TDMCorba1-NomPrenom.pdf **sur moodle chez TOUTES les personnes du binôme.** Ce CR contiendra les informations que vous jugerez nécessaires.
- Votre CR sera disponible pour vous lors de l'examen machine.