# M'sila University, Department of Computer Science,

# ISIL

**COURSE: DISTRIBUTED INFORMATION SYSTEMS** **DR. R. BENTRCIA**

**TP 2: Distributed Information Systems (Solution)**

## Exercise 1:

### // RMIInterface.java

```
import java.rmi.Remote;
import java.rmi.RemoteException;


public interface RMIInterface extends Remote {
	public String helloTo(String name) throws RemoteException;
}
```

### // ClientOperation.java

```
import java.net.MalformedURLException;
import java.rmi.Naming;
import java.rmi.NotBoundException;
import java.rmi.RemoteException;
import javax.swing.JOptionPane;


public class ClientOperation {
	private static RMIInterface look_up;
	public static void main(String[] args) throws MalformedURLException, RemoteException,    NotBoundException {
		look_up = (RMIInterface) Naming.lookup("//localhost/MyServer");
		String txt = JOptionPane.showInputDialog("What is your name?");
		String response = look_up.helloTo(txt);
		JOptionPane.showMessageDialog(null, response);
	}
}
```

**// ServerOperation.java**

```
import java.rmi.Naming;
import java.rmi.RemoteException;
import java.rmi.server.UnicastRemoteObject;


public class ServerOperation extends UnicastRemoteObject implements RMIInterface{
        protected ServerOperation() throws RemoteException {}
        public String helloTo(String name) throws RemoteException{
                System.out.println(name + " is trying to contact!");
                return "Server says hello to " + name;
        }
        public static void main(String[] args){
          try {
             Naming.rebind("//localhost/MyServer", new ServerOperation());
             System.out.println("Server ready");
           } catch (Exception e) {
             System.err.println("Server exception: " + e.toString());
            e.printStackTrace();
           }
      }
}
```

## Exercise 2:

### // AddInterface.java

```
import java.rmi.Remote;
import java.rmi.RemoteException;


//The Remote Interface

public interface AddInterface extends Remote{//Extend the Remote interface
 String add(int x, int y) throws RemoteException; //Declare the RemoteException
}
```

### // MyClient.java

```
import java.net.MalformedURLException;

import java.rmi.Naming;

import java.rmi.NotBoundException;

import java.rmi.RemoteException;


//The Client Application

public class MyClient{

   public static void main(String arg[]){

      int number1 = 10;

      int number2 = 15;

    try {  //It returns the reference of the remote object obj in registry

         AddInterface obj = (AddInterface) Naming.lookup("//localhost/MyServerTool");

         System.out.println(obj.add(number1, number2)); //Invoking the remote method on this object

      }

      catch (Exception e){

        System.out.println("MyClient exception: " + e.getMessage());

        e.printStackTrace();

      }

   }

}
```

```
// MyServer.java
import java.rmi.Naming;
import java.rmi.RemoteException;
import java.rmi.server.UnicastRemoteObject;

//The Server Application
public class MyServer extends UnicastRemoteObject implements AddInterface{
  public MyServer() throws RemoteException {} // Define a constructor that declares RemoteException
  public String add(int number1, int number2) {//Implement the remote method add
            int sum=0;
            sum = number1+number2;
            return ("The sum is: " + sum);
            //return sum;
  }
  public static void main(String args[]){
    try{
      MyServer obj1 = new MyServer(); //Create an instance of the remote object
      // Bind this object instance to the name "MyServerTool" in the rmiregistry
      Naming.rebind("//localhost/MyServerTool", obj1);
      System.out.println("The server is ready!");
    }
    catch (Exception e){
      System.out.println("MyServer err: " + e.getMessage());
      e.printStackTrace();
    }
  }
}
```