```
package client;

import interfaces.*;
import java.rmi.*;

/**
 * <p>Title: </p>
 * <p>Description: </p>
 * <p>Copyright: Copyright (c) 2002</p>
 * <p>Company: </p>
 * @author unascribed
 * @version 1.0
 */

class PrintHello implements Task {
    public PrintHello() {
    }

    public Object run() {
        System.out.println("Hello world !");

        return null;
    }
}


public class Client {
    public final static String name = "//127.0.0.1/Executor";

    public Client() {

    }

    public static void main(String[] s) {
        try {
            Executor server = (Executor) Naming.lookup(name);
            server.execute(new PrintHello());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
package server;
import interfaces.*;

import java.rmi.*;
import java.rmi.server.*;


public class Server extends UnicastRemoteObject
                    implements Executor
{

    public final static String name = "//127.0.0.1/Executor";

    public Server() throws RemoteException {
        super();
    }


    public static void main(String[] s) {
        try {
            Executor server = new Server();
            Naming.rebind(name, server);
            System.out.println("Bound");
        } catch (Exception e) {
            System.err.println("Exception: " + e.getMessage());
            e.printStackTrace();
        }
    }


    //-------------------------------------------------------------------
    // Implementation of the Executor interface

    public Object execute(Task task) throws RemoteException {
        System.out.println("Server received command. Starting calculations");
        Object ret = task.run();
        System.out.println("Server ended calculation. Returning value");
        return ret;
    }

}
```

```
package interfaces;

import java.rmi.*;

public interface Executor extends Remote {
  public Object execute(Task task) throws RemoteException;
}
```

```
package interfaces;

import java.rmi.*;

public interface Task extends java.io.Serializable {
    Object run();
}
```